# DOCUMENT RESUME

ED 201 410 PS 012 175

| | |
|---|---|
| TITLE | Exploring with Computers in Vikingland. AEDS-81 Convention Proceedings (Minneapolis, Minnesota, May 5-8, 1981). |
| INSTITUTION | Association for Educational Data Systems, Washington, D.C. |
| PUB DATE | May 81 |
| NOTE | 319p.; Figures appearing on pages 21, 23-28, and 273 of original document may not reproduce well due to small print size. |
| AVAILABLE FROM | Association for Educational Data Systems, 1201 Sixteenth Street, N.W., Washington, DC 20036 (Paper, $15.00). |
| EDRS PRICE | MF01/PC13 Plus Postage. |
| DESCRIPTORS | *Computer Assisted Instruction; *Computer Managed Instruction; *Computer Oriented Programs; Computers; *Computer Science Education; Higher Education; Microcomputers; Music Education; Programing Languages; Special Education |
| IDENTIFIERS | Association For Educational Data Systems; Computer Literacy; Videodiscs |

ABSTRACT

This publication presents a summary of and index to the presentations given at the Association for Educational Data Systems (AEDS) Convention held in Minneapolis, Minnesota, during May 5-8, 1981. Summarized are 66 short papers that cover a variety of educational computing activities and projects completed by educational institutions, organizations, and individuals across the United States. To facilitate use of the publication, papers are indexed by author and by the following topics: administrative computing, computer awareness/literacy, computer languages, computers and music education, computers and special education, computers and the media center, computers in higher education strand, instructional computing, instructional design, instructional management, software clearinghouses/evaluation, and videodisc. Lists of AEDS' 1981 convention team, regional workshop coordinators, directors, affiliate group presidents and sustaining members are also included. (Author/MP)

***********************************************************************
* Reproductions supplied by EDRS are the best that can be made
* from the original document.
***********************************************************************

ED201410

# AEDS-81

U.S. DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

# CONVENTION

# PROCEEDINGS

PS012175

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
AEDS

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Minneapolis, MN
May 5-8,1981

## Exploring With Computers
## In
## VIKINGLAND

Additional copies available for
$15.00, prepaid, from

Association for Educational Data Systems
1201 Sixteenth Street, N.W.
Washington, D.C. 20036

Library of Congress Catalog Card Number ISSN 0147-9296

Copyright (c) 1981, Association for Educational Data Systems

# P R E F A C E

As you read the contents of this publication, please realize that most of the activities reported upon herein have already taken place, and have produced positive changes for educational computing. Your purpose in reading this publication is not to see what others have done, but to learn what you can do for the promotion of educational computing. Thinking along the lines of how you can replicate, or adapt what others have done will assist you in the short and long term. Incorporating bits and bytes from a variety of computing activities and projects completed by others will allow you to more effectively and efficiently build and implement computing services models designed to meet your own needs. It is with this intent, the improvement of educational computing, that the AEDS organization prepares this publication for distribution.

The planners, preparers, and presentors of AEDS-81 labored to produce a complete convention which would produce intercourse and interchange between individuals regards educational computing. The contents of this publication summarizes the presentations given at the AEDS-81 Convention, held in Minneapolis, Minnesota during May 5-8, 1981. Two indices (author and topic) should facilitate quick reference to all of the papers submitted in time for publication.

Minnesota educational computing organizations, institutions, and individuals are proud to have been able to host the AEDS-81 Convention. The AEDS organization thanks you for your support of educational computing and hopes that the information in this publication will assist all of us in improving our profession.

Publication layout and preparation by Teri Leonard, MECC Administrative Secretary, Instructional Services Division.

# TABLE OF CONTENTS

page

PREFACE iii

TABLE OF CONTENTS iv

1981 CONVENTION TEAM v

REGIONAL WORKSHOP COORDINATORS vi

AEDS BOARD OF DIRECTORS vii

AFFILIATE GROUP PRESIDENTS viii

SUSTAINING MEMBERS OF AEDS xii

INDEX OF PAPERS BY AUTHOR xiii

PAPERS 1

INDEX OF PAPERS BY TOPIC 295

# 1981 AEDS CONVENTION TEAM

## CONVENTION CO-CHAIRMEN

Ken Brumbaugh
MECC
2520 Broadway Drive
St. Paul, Minnesota 55113

Dale Schneiderhan
MECC
2520 Broadway Drive
St. Paul, Minnesota 55113

## PROGRAM

Robert D. Nelson
Richfield Public Schools
7001 Harriet Avenue South
Richfield, Minnesota 55423

Sue Gerk Talley
TIES
1925 W. County Road B2
Roseville, Minnesota 55113

## PUBLICITY

Darlyne Pogose
MECC
2520 Broadway Drive
St. Paul, Minnesota 55113

Jenni Wilken
MECC
2520 Broadway Drive
St. Paul, Minnesota 55113

## EXHIBITS

Jim Altermatt
Tele-Terminals
7216 Boone Avenue North
Brooklyn Park, MN 55428

Doug LaChance
MECC
2520 Broadway Drive
St. Paul, Minnesota 55113

## SPECIAL ACTIVITIES

Vi Hakola
Anoka-Hennepin Public School
11299 Hanson Blvd. NW
Coon Rapids, MN 55433

Carolyn Meinhardt
San Diego City Schools
4100 Normal Street
San Diego, California 92103

## REGISTRATION AND FINANCE

Jerry Foecke
ESV Region III
570 First St. SE
St. Cloud, Minnesota 56301

Russell E. Weitz
TIES
1925 W. County Road B2
Roseville, Minnesota 55113

## FACILITIES AND EQUIPMENT

Marien A. Haley
TIES
1925 West County Road B2
Roseville, Minnesota 55113

Rich Rowe
Lakeville Public Schools
8670 210 Street West
Lakeville, MN 55044

## CONVENTION OFFICE

Darlene Crawford
Rosemount Public Schools
14445 Diamond Path West
Rosemount, Minnesota 55068

Rosemary Schneiderhan
Roseville Public Schools
1251 W. County Road B2
Roseville, Minnesota 55113

## MESSAGE CENTER

Chris Papadoulis
6774 E. Shadow Lane
Lino Lakes, Minnesota 55110

Del Schroeder
Box 91
Stillwater, Minnesota 55082

## SPECIAL SESSIONS

(VENDORS)

Don Rawitsch
MECC
2520 Broadway Drive
St. Paul, Minnesota 55113

(COUNCIL ON QUALITY EDUCATION)
Marge Kosel
MECC
2520 Broadway Drive
St. Paul, Minnesota 55113

## REGIONAL WORKSHOP COORDINATORS

Don Holznagel
NWREL(MicroSIFT)
300 SW Sixth Avenue
Portland, Oregon 97204

Ken Brumbaugh
MECC
2520 Broadway Drive
St. Paul, MN 55113

Gene Collins
Jefferson Co. Schools
809 Quail Street
Lakewood, CO 80215

Don Costello
University of Nebraska
Lincoln, Nebraska, 60812

Ralph Lee
University of Missouri
Computer Center
Rolla, MO 65401

## AEDS BOARD OF DIRECTORS
## 1980-81

**President Winston Addis**
Supt. of Schools
Mt. Vernon Community Schools
Mt. Vernon, Iowa 52314
319/895-8845

**Past President Don Holznagel**
MicroSIFT,NWREL
300 SW 6th Avenue
Portland, Oregon 97204
503/248-6800

**Treasurer Steve Raucher (1982)**
Montgomery Co. Public Schools
850 Hungerford Drive
Rockville, Maryland 20850
301/279-3581

**Bobby Brown (1981)**
Computer Center
University of Iowa
Iowa City, Iowa 52242
319/353-3170

**Jerome Foecke (ex-officio)**
ESV Region III
570 First St. SE
St. Cloud, Minnesota 56301
612/255-9440

**Dale LaFrenz (1981)**
Scott, Foresman Co.
1900 E. Lake Avenue
Glenview, Illinois 60025
312/729-3000

**Carolyn Meinhardt (1982)**
San Diego City Schools
Education Center, Room 1017
4100 Normal Street
San Diego, California 92103
714/293-8379

**Russell Weitz (ex-officio)**
**Functional Affiliate Rep.**
TIES
1925 W. County Road B2
St. Paul, Minnesota 55113
612/633-9110

**President-Elect Dennis Spuck**
Associate Professor and Director
Education Research Center
AED/FH University of Houston
Houston, Texas 77004
713/749-7281

**Secretary James Gunderson (1983)**
Seattle Public Schools
816 Fourth Avenue North
Seattle, Washington 98109

**Arthur Daniels, Jr. (1983)**
RAMS Cooperative
31201 Dorchester
Madison Heights, Michigan 48071
313/585-7530

**Judith Edwards (ex-officio)**
NWREL
300 SW Sixth Avenue
Portland, Oregon 97204
503/248-6800

**D. Thomas King (1982)**
St. Paul Public Schools
360 Colborne Street
St. Paul, Minnesota 55102
612/298-5867

**A.V. Majors (ex-officio)**
**Geographic Affiliate Rep.**
Ft. Worth Public Schools
3210 W. Lancaster
Ft. Worth, Texas 76107
817/336-8691

**Rick Meyer (1983)**
Maricopa Community Colleges
3910 E. Washington Street
Phoenix, Arizona 85043
602/244-8355

**Shirley Easterwood**
**Executive Secretary**
AEDS
1201 16th Street NW
Washington, D.C. 20036
202/833-4100

## AFFILIATE GROUP PRESIDENTS

**Functional:**

INFORMATION PROCESSING
ADMINISTRATORS OF LARGE SCHOOL SYSTEMS

Roy Ducharm (5/80-5/81)
Prince Georges County
Public Schools
Upper Marlboro, Maryland 20870
301/952-4378

Permanent Contact

Arthur Daniels, Jr.
RAMS Cooperative
31201 Dorchester
Madison Heights, Michigan 48071
313/585-7530

**Geographic:**

ALBERTA

H.J. Hallworth (7/80-6/81)
Faculty of Education
University of Calgary
Calgary, Alberta, Canada
403/284-6493

ARIZONA

James Lee (5/80-4/82)
American Graduate School Intermediate Management
Thunderbird Campus
Glendale, Arizona 85306
602/978-5401

CALIFOIRNIA EDUCATIONAL DATA PROCESSING ASSOCIATION

Al Grossman (11/80-11/81)
Administrator, Office of Information Systems
San Mateo County Office of Education
333 Main Street
Redwood City, California 94063
415/364-5600 x2554

FLORIDA

F. Ronald McCord (6/80 - 11/81)
Manatee Jr. College
P.O. Box 1849
Brandenton, Florida 33506
813/755-1511

Permanent Contact

Randall Johns
Director Data Processing
202 E. Commercial Street
Sanford, Florida 32721
305/322-1252

ILLINOIS

Lyle Smith (11/80 - 10/81)
Computer Science Dept. PM 556
Northern Illinois University
DeKalb, Illinois 60115
815/753-0378

## IOWA

Charles Ruebling (11/80 - 10/81)
Mason City Community Schools
120 East State Street
Mason City, Iowa 50401
515/421-6217

Permanent Contact

Winston Addis
Mt. Vernon Community Schools
Mt. Vernon, Iowa 52314
319/895-8845

## MANITOBA

Mark Bishop (4/80 - 5/81)
Red River Community College
2055 Notre Dame Avenue
Winnipeg, Manitoba R3H 0J9
204/632-2278

Permanent Contact

Edward Boorsma
Winnipeg School Distict #1
1577 Wall Street East
Winnipeg, Manitoba R3E 2S5
204/775-0231

## MARYLAND

Frank M. White (4/81 - 3/82)
Dept. of Computer DP Tech.
Catonsville Community College
Baltimore, Maryland 21228
301/461-9321

## MICHIGAN

Nick Colvos (10/80 - 9/81)
Lansing Public Schools
519 W. Kalamazoo
Lansing, Michigan 48933
517/374-4791

Permanent Contact

Arthur Daniels, Jr.
RAMS Cooperative
31201 Dorchester
Madison Heights, Michigan 48071
313/585-7530

## MID-AMERICA

Wilbur C. Dorsey (6/80 - 6/81)
Wichita Public Schools
428 S. Broadway
Wichita, Kansas 67202
316/268-7768

## MID-SOUTH

Dr. Benjamin Showalter (2/80 - 2/82)
Educ. Research and Evaluation
University of Alabama - Birmingham
University Station, Alabama 35294
205/934-2357

Permanent Contact

Dr. Walter M. Mathews
School of Education
University of Mississippi
University, Mississippi 38677
601/232-7195

## MINNESOTA

Ken Brumbaugh
MECC
2520 Broadway Drive
St. Paul, Minnesota 55113
612/376-1105

MISSOURI

Harry Thompson
1916 Elm
St. Charles, Missouri 63301

**Permanent Contact**

Ralph Lee
P.O. Box 951
Rolla, Missouri 65401

NEBRASKA

Dirk Dierkhising (5/80 - 5/81)
Omaha Public Schools
4469 Farnam
Omaha, Nebraska 48153
402/554-6192

NEW JERSEY

James Megletti
Teaneck Public Schools
1 West Forest Avenue
Teaneck, New Jersey 07666
201/837-2232

**Permanent Contact**

Irvin Gaydos
State Dept. of Education
225 West State
Trenton, New Jersey 08601
609/292-6037

NEW YORK

Charles Adams (10/80-10/81)
Onandaga BOCES
6820 Thompson Road
Syracuse, New York 13211
315/437-1631

**Permanent Contact**

NYSAEDS
P.O. Box 164
Glenwood, New York 14069

NORTH CAROLINA

Dr. John Beegle
Western Carolina University
Accounting and Info. Systems
Cullowhee, North Carolina 28723
704/227-7403

OREGON

Paul O'Driscoll (1/81-12/81)
Linn. Benton ESD
905 E. 4th Street
Albany, Oregon 97321
503/967-4381

TEXAS

Philip Gensler (10/80-10/81)
Dept. of CIS
W. Texas State University
Canyon, Texas 79016

VIRGINIA

Roger Higgs (6/80-6/81)
Coord. Admin. Data Systems
College of William and Mary
Williamsburg, Virginia 23185
804/427-4477

Permanent Contact

James Augustine
Norfolk City Public Schools
800 E. City Hall Avenue
Norfolk, Virginia 23510
804/441-2458

WISCONSIN

John C. Bernard (5/80-6/81)
Kenosha Unified SD #1
625 - 52nd Street
Kenosha, Wisconsin 53141
414/656-6347

Permanent Contact

Earl Chandler
Dept. of Public Instruction
126 Langdon Street
Madison, Wisconsin 53702
608/267-9206

## SUSTAINING MEMBERS OF AEDS

Management Science America, Atlanta Georgia

MICRO Power And Light, Dallas Texas

National Computer Systems, Minneapolis Minnesota

Prime Computer, Wellesly Massachusetts

Time Share Corporation, Hanover New Hampshire

Westinghouse Learning Corporation, Iowa City Iowa

## INDEX OF PAPERS BY AUTHOR

| | | Page |
|---|---|---|
| Alspaugh, John W. | Computer Assisted School Bus Route Construction With A Microcomputer | 1 |
| Bailey, T.E. | Pseudocode And Constructs For Structured Programming In FORTRAN (1966 Standard) And BASIC | 3 |
| Barber, Frank | People Make The Difference | 9 |
| Bell, Fred | Computer Literacy Courses And Computer Applications In Education: What Should They Be? | 12 |
| Berry, Robert I. | Zero Base Budgeting - A Management System Approach For Educational Institutions | 18 |
| Blombach, Ann K.,<br>Murphy, Barbara A. | Music Ear Training At Ohio State: The Phoenix/Orion Project | 29 |
| Boe, Thomas F. | Effective Integration Of Computer Methods In The Science Classroom | 34 |
| Borry Hausmann, Linda | Effective Feedback In Computer Assisted Instruction | 38 |
| Brebner, Ann,<br>Hallworth, H.J.,<br>Woetowich, H.M.<br>Mah, S.N.,<br>Huang, H.C. | Results And Implications of Some Recent Research on CAI in Mathematics | 42 |
| Browne, Diane<br>Corens, Ken | Microcomputers In Cooperative Groups | 47 |
| Carrison, Dale K. | MSUS/PALS: Planning For The Future | 48 |
| Carruth, Gayden F. | Position Paper: Common Information Base For Regional and Local Decision Making | 52 |
| Cogdill, Linda J.,<br>Goldberg, Albert L. | An Information System For Managing HEAD START | 56 |
| Dalgaard, Bruce R.,<br>Askalani, Mohamed,<br>Thompson, Paul | Computer Assisted Instruction in Economics: Development, Results, and Challenges | 59 |

| | | |
|---|---|---|
| D'Angelo, John | TI LOGO: A Turtle Of A Different Color | 65 |
| Daniels, Arthur W. | The Development and Operation Of A Programming Cooperative | 69 |
| Delf, Robert M. | Application Software: Buyer's Guidelines | 74 |
| Dellow, Donald A.,<br>Poole, Lawrence H.,<br>Clark, Forrest | The Role of Microcomputers In Community College Business and Secretarial Science Programs | 80 |
| Donoian, George | A Paradigm For Computer-Based Instruction: Renaissance of the Affective Domain | 84 |
| Erting, Lynne,<br>Mackall, Phil | Academic and Non-Academic Benefits From Using PLATO - A Classroom Teacher's View | 89 |
| Garvey, Mary J. | Use of CAI In A Mainstreamed Hearing-Impaired Program | 97 |
| Gockel, James | Large Time-Sharing Microcomputers In Secondary Education | 100 |
| Gross, Dorothy,<br>Griffing, Wendy | Implementation and Evaluation Of A Computer-Assisted Course in Musical Aural Skills | 104 |
| Hadduck, Charles G. | An Employment Applications Information System | 107 |
| Hamilton, Gregory C. | A Microcomputer Managed Instructional System | 112 |
| Hartman, Janet,<br>Garnett, Dianne | A Structured Approach To The Development Of Computer Assisted Instruction (CAI) | 119 |
| Hassell, Johnette | A Paradigm For Developing A Computer Science Program | 125 |
| Haugo, John E. | Management Applications Of The Microcomputer: Promises And Pitfalls | 129 |
| Heck, Bill | Microcomputers In Education | 133 |
| Holznagel, Donald C. | A Clearinghouse For Educational Applications Of Microcomputers | 136 |
| Hurley, Paul,<br>Davis, Virginia | Lifelong Learning And The Post Industrial Revolution | 139 |
| Isaacs, G.L. | Realities Of Transportable Micro-Courseware With Graphics Emphasis | 145 |

| | | |
|---|---|---|
| Judd, Dorothy | Sources For and Evaluation Of K-12 Microcomputer Software | 150 |
| Kahn, Phyllis | The State's Role In Educational Computing: View From The Legislature | 157 |
| Kehrberg, Kent T.,<br>Pollak, Richard A. | Low Cost Computer/Videodisc Learning Stations | 159 |
| King, Gary,<br>Kobylarz, Linda | The GIS Options Program: A Dynamic Approach To Providing Localized Career Information | 162 |
| Koehler, Tommas J.,<br>Ossler, Anna C.,<br>Raucher, Stephen M. | Computerized Support Systems For Individual Educational Plans (I.E.P.) | 165 |
| LaChance, Doug,<br>Stokka, Keith | Using A Microcomputer With A Large Data Base | 169 |
| Lathrop, Ann | SOFTSWAP: The Evaluation And Dissemination Of Non-Commercial Teacher Created Software | 173 |
| Lindsay, E. Jay | Learning By Computer Practice | 176 |
| Loewen, Duane | Microcomputers In Teaching Handicapped Students | 182 |
| Lorton, Paul,<br>Muscat, Eugene J. | Administrative Use For The Thimble: Micro-computer Based Attendance System | 185 |
| Majors, A.V.,<br>Westerfeld, Lloyd | Student Intelligent Terminal System | 190 |
| Mayer, Herbert C. | Information Systems Management | 195 |
| Molloy, Lawrence,<br>Herstein Ed | A Computer Demonstration Room For Educators | 203 |
| Mosow, David K. | The Microcomputer and Management of the Time Bound Educational Program | 206 |
| Nelson, Robert W.,<br>Kepner, Henry S. | Computers In Education - Meeting The Challenge | 210 |
| Postman, Robert D. | ROBOT PROBE: An Introduction To Programming | 214 |
| Raucher, Stephen M.,<br>Koehler, Tommas J. | Management Operations Review and Evaluation (MORE) of a School District Computer Services Department | 217 |
| Rawitsch, Don G. | The Organization Of User Services For Instructional Computing | 222 |

| | | |
|---|---|---|
| Rice, Jean,<br>O'Connor, Sandy | Computer Literacy In The Elementary Classroom | 229 |
| Rich, Barbara | Implementation Of A College-Wide Computer Literacy Program: A Working Model | 234 |
| Rizza, Peter J. | For Computer-Based Education To Matter It Needs Critical Mass | 237 |
| Roblyer, M.D. | Instructional Design vs. Authoring Of Courseware: Some Key Differences | 243 |
| Roll, Peter | Long-Range Planning For Information Services At Large Universities | 248 |
| Settlemire, James R. | Distributed Systems In The School District Environment | 253 |
| Sinofsky, Esther R.,<br>Knirk, Frederick G. | A Systematic Procedure For Problem Analysis And Proposal Development | 257 |
| Uffelman, Robert L.,<br>Boas, Edward E. | Delta Computer-Managed Instruction System | 260 |
| Van Dusseldorp, Ralph | Alaska On-Line Resource Information Bank | 263 |
| Wachtel, Edward E. | Computer Science Summer School Offering - A Place To Start | 267 |
| Wachtel, Edward E.,<br>Rankin, H. Clair,<br>Obrock, Madeleine,<br>Burns, Evelyn H.,<br>Bazeley, John | A Five Year Time Line For The Integration Of Computers Into The Shaker Heights City Schools | 270 |
| Weitz, Russell E. | Word Processing Evaluation and Selection | 275 |
| Whitney, Richard A.,<br>Hofmeister, Alan M. | MONITOR: A Computer Based Management Information System for Special Education | 279 |
| Wiederanders, Don E.,<br>Schwandt, Lynn C. | Microcomputers: Integration Into The Curriculum | 287 |
| Wilde, Jane H. | The Computer In Developmental Education | 290 |

# COMPUTER ASSISTED SCHOOL BUS ROUTE CONSTRUCTION WITH A MICROCOMPUTER

John W. Alspaugh

College of Education
University of Missouri-Columbia

The application employs the distance-saved concept for the construction of school bus routes requiring a minimum of buses and miles traveled. The location of each stop is recorded on a one inch-per-mile highway map. The inter-stop distances for possible connector stops and the distance from school to each stop are measured. The data for the program are the measured distances and the number of students at each stop. An Apple II plus computer with 48K and one disk is used to construct routes that may be serviced with available buses under time constraints set within the school district.

With inflation and the increasing cost of buses and gasoline, school districts are confronted with rapidly increasing transportation costs. One of the approaches used to reduce transportation costs is to alter the bus routes to reduce the number of bus routes and miles traveled. A second approach, applicable only in districts with high population densities, is to alter the starting times of individual schools and use the buses for multiple routes. A combination of the two approaches appears to be most effective in reducing costs in high population areas. A reduction in the number of buses is usually associated with a reduction in the miles traveled, except in some districts with multiple routes for each bus. A reduction in the number of buses tends to lower costs faster than reducing the miles traveled.

Development of a good set of routes is a very tedious and time consuming task. As a result, most districts simply modify their existing routes year after year rather than develop new ones. After a few years this modification process leads to inefficient routes.

## General Procedure

Researchers who have studied the route construction process generally recommend a distance saved approach. The first step is to plot the stops on a large scale road map. The stops are assigned identification numbers 1, 2, 3, etc. Then the distance from each stop to school, and from each stop to all possible connector stops on potential routes, are measured to the nearest tenth of a mile. To reduce the effort required to measure the map, the inter-stop distances are measured from each stop to only higher numbered stops and the symmetric distances are filled in by the computer. Accurate measurement is critical to the route construction process. The following information is keyed into the computer for each stop: (1) number of students, (2) distance to school, and (3) inter-stop connector stop numbers and inter-stop distances.

The actual computations performed in the route construction process are numerous and complex. The general procedure is as follows:

1. As a starting point, assume that each stop represents a separate route.
2. Search all of the available inter-stop connectors to find which two stops may be joined to form a route with two stops that will save the most distance traveled.
3. Keep a running total on the number of stops, the number of students, and the length of the route.
4. If the route is too long then go to step 7, else if the bus is loaded then go to step 8.
5. Find the stop that may be added to either end of the route and save the maximum traveled distance. Add it to the route.
6. Go to step 3.
7. Add stops to the route that are between school and the first stop on the route, and between the last stop on the route and school until the bus is full.
8. Record the route and go to step 2. Continue until all stops have been assigned to a route.

The computer is used to construct two or three alternate sets of school bus routes for a district. Individuals within the district, responsible for the transportation system, may then select from these alternatives the set of routes that is judged to best meet the needs and concerns of the district.

## Computer Hardware Requirements

The system is operational on an Apple II plus computer with 48K and one disk drive. A printer is optional. There could be some advantage in using multiple disk drives and more core for large school districts.

## Files in the Data Set

The system employs four files. The SCHOOL file is a sequential file with the following entries: school name, number of stops, number of constructed routes, maximum number of students-per-stop, and maximum distance from any stop to school. The STOP file is a random access file with one record-per-stop. Each record contains the number of students, distance to school, route number if assigned to a route, and up to twelve inter-stop connector stop numbers and distances. The ROUTE file is a random access file with one record for each constructed route. Each record contains the number of students, number of stops, length of the route, and stops assigned to the route. The ROUTE INDICATOR file is a sequential file containing a one dimensional array. Each array element contains the route number that the corresponding stop is assigned to.

During the route construction process the SCHOOL and ROUTE INDICATOR files are in core. The individual records of the STOP and ROUTE files are read in individually as they are needed.

The data set may accommodate about 450 stops on a single disk. Large school districts with more stops would need to subdivide the stops into areas or attendance centers. The program could be changed to use multiple disk drives.

## Computer Programs in the System

The system consists of four programs. The DATA program is used to set up the SCHOOL and STOP files. The COPY program is used to make backup copies of the SCHOOL and STOP files. The SYMMETRIC program is used to fill in the symmetric inter-stop distances in the STOP file and reproduce the SCHOOL and STOP files. The CONSTRUCT program creates the ROUTE and ROUTE INDICATOR files as it constructs the routes.

The CONSTRUCT program uses a heuristic approach to the route construction process. It employs a series of commands to build the routes one stop at a time. The commands to operate on the data set are as follows:

```
FIND START
     STOP
     B STOP
     E STOP

ENTER SCHOOL
      ROUTE
      B STOP
      E STOP
      ROUTE NUMBERS
      ROUTE INDICATOR
      CONNECTOR

DELETE ROUTE
       B STOP
       E STOP
       STOP
       CONNECTOR

DISPLAY STOP
        SCHOOL
        UNASSIGNED STOPS
        INACTIVE STOPS
        SUMMARY

B TO SCHOOL
E TO SCHOOL

RECORD

CHECK STOPS ON ROUTES

PRINT
QUIT
END
```

These commands are used in an interactive environment with the computer. Each route is considered as a loop from school through the stops and back to school. The first stop is called B stop and the last stop is called E stop.

## Characteristics of Optimum Routes

The computer is used to construct alternate sets of routes. The school uses the set of routes that best meet the unique needs of the district.

"Ideal" routes tend to be shaped like petals on a daisy. There is a minimum of turn arounds and back tracking. The routes tend to be non-overlapping. The buses are as large as possible and filled to capacity, thus yielding a minimum number of routes and miles traveled.

## References

Clarke, G., and J. W. Wright, "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points", Operations Research 12:568-81, May-August 1964

Dantzig, G. B., and J. H. Ramser, "The Truck Dispatching Problem", Management Science 6:89-91, October 1959

Gillett, Billy E., and Leland R. Miller, "A Heuristic Algorithm for the Vehicle Dispatch Problem", Operations Research 22:340-9, March-April 1974

Robbins, J. A., and W. C. Turner, "No Computer? Try Wedging Bus Routes", Nations Schools and Colleges 2:44-45, May 1975

Schruben, Leonard W., and Ruth E. Clifton, "The Lockset Method of Sequential Programming Applied to Routing Delivery and Pickup Trucks", American Journal of Agricultural Economics 50:854-68, November 1968

# PSEUDOCODE AND CONSTRUCTS FOR STRUCTURED PROGRAMMING IN FORTRAN (1966 STANDARD) AND BASIC

T. E. Bailey

Department of Computing and Information Sciences
Oklahoma State University
Stillwater, Oklahoma 74078

Commonly used structured constructs (such as *if-then-else* and *do while*) applied to BASIC and 1966 Standard Fortran are at best awkward. The widespread use of these languages in minicomputer applications, education (often as a "teaching" language), and the innumerable other uses of Fortran necessitates selection and repetition constructs easily adaptable to them, yet compatible with the commonly used constructs. A simple structured pseudocode with keywords *test* for all types of selection (2-way, 3-way, and *n*-way) and *loop* for all cases of looping has been developed, which can be translated easily and naturally into BASIC or Fortran.

## Introduction

The use of structured concepts in program design is inconvenient when the computer language used to implement programs is Fortran (1966 Standard) or BASIC. While it is recognized that some versions of BASIC and Fortran, such as WATFIV, do contain constructs for *if-then-else* and *do while* and that Fortran 77 provides an *if-then-else* construct, many programmers must learn/teach and otherwise use versions of Fortran or BASIC which do not. This paper is addressed to those who must use Standard Basic Fortran, Standard Fortran (1966), and "unstructured" versions of BASIC--those versions of the languages which do not include special constructs for selection (*if-then-else*) and for repetition (*do while* or *do until*). The words *Fortran* and *BASIC* in the remainder of this paper therefore refer to these versions.

## Statement of the Problem

At Oklahoma State University about 1800 students enroll in the beginning programming course each year. They range from freshmen to graduate students, with backgrounds in nearly all majors. Standard Fortran is used as the programming language in this course, and also in the second course (which is primarily for computer science majors), for a number of reasons, not the least of which is that OSU is a land grant university with emphasis in the various sciences, agriculture, and engineering. For three years we have been trying to incorporate into the beginning course structured program design techniques with pseudocode, similar to those described by Shelly and Cashman (1). Requirements for the pseudocode included the following: (1) wide applicability--useful for implementation in Pascal, PL/1, COBOL, BASIC, Algol, and Fortran; (2) simplicity--English-like and natural, with few constraints; and (3) availability of commonly used constructs for selection and repetition. The objective of this approach was to improve learning in the skills of programming, to narrow the gap between the first and second programming courses, and to prepare students for programming in other languages, particularly PL/1, which is the principal language used in upper level courses.

Two problems quickly became evident: the lack of suitable textbooks and the apparent lack (for the beginning student, at least) of any relationship between Fortran and structured constructs.

Though some authors, such as Cole (2), Hull and Day (3), and Chirlian (4), have made attempts at "structured" Fortran, consistency and clarity often seem to be lacking. In Cole's book, for example, structured concepts are placed in a chapter near the end of the book. Hull and Day, on the other hand, introduce structured concepts with pseudocode early in their book, but seem not to follow them consistently through the rest of the book. Chirlian's book, while handling structured concepts well, does not use pseudocode. We decided, therefore, to produce a new supplementary textbook dealing only with structured design concepts through the use of a rather unconstrained and natural pseudocode, which is now in preparation and testing.

The second problem, devising structured constructs for Fortran, has been addressed by the development of a simple, natural pseudocode that can be translated into Fortran code in a direct, straightforward manner. This pseudocode turned out to be a special case of a more general pseudocode which contains *if-then-else* and *loop while* constructs. The *test* construct handles all types of selection, while the *loop* construct applies to all cases of looping.

### The *Test* Construct for Selection

Selection involves, in the simplest case, the comparison of two pieces of information and the selection of one of two alternative actions, depending upon the outcome of the comparison. Selection may be generalized to include the comparison of more than two pieces of information and the selection of one of several alternative actions.

In Fortran, there are three types of selection: the logical IF (two-way selection), the arithmetic IF (three-way selection), and the computed and assigned GO TOs (*n*-way selection). The conventional construct for two-way selection is the *if-then-else*; but Fortran's two-way selection statement, the logical IF, does not match the general case of the *if-then-else*. Consider, for example, the following program segment, stated in pseudocode:

```
If salary is greater than $15000, then
     tax is 25%.
Else
     tax is 20%.
End if.
```

If we attempt to implement this in Fortran, we could get either of the following program segments:

```
C --- IF SALARY IS GREATER THAN $15000,
      IF(SALARY.LE.15000.)GO TO 5
C ---    THEN TAX IS 25%.
            TAX = 0.25 * SALARY
            GO TO 6
C ---    ELSE TAX IS 20%.
    5       TAX = 0.20 * SALARY
    6 CONTINUE
C --- END IF
```

```
C --- IF SALARY IS GREATER THAN $15000,
      IF(.NOT.(SALARY.GT.15000.))GO TO 5
C ---    THEN TAX IS 25%.
            TAX = 0.25 * SALARY
            GO TO 6
C ---    ELSE TAX IS 20%.
    5       TAX = 0.20 * SALARY
    6 CONTINUE
C --- END IF
```

Both of the IF statements are potentially confusing to the beginning student, since the sense of the test is exactly opposite what is stated in the pseudocode, because of the way the logical IF operates.

The *test* construct has been devised to overcome this problem. While it departs somewhat from the *if-then-else* construct, it preserves the sense of the test and at the same time matches the structure of the logical IF. The tax program can be stated with the *test* construct in pseudocode as follows:

```
Test salary greater than $15000.
     False:  tax is 20%.
     True:   tax is 25%.
End test.
```

The Fortran program then becomes

```
C --- TEST SALARY GREATER THAN $15000.
      IF(SALARY.GT.15000.)GO TO 5
C ---    FALSE.  TAX IS 20%.
            TAX = 0.20 * SALARY
            GO TO 6
C ---    TRUE.  TAX IS 25%.
    5       TAX = 0.25 * SALARY
    6 CONTINUE
C --- END TEST
```

The general form of the *test* construct is shown in Figure 1 with its corresponding Fortran code. A CONTINUE statement is used as the exit from the construct. If the selection has a null alternative (one containing no statements), then the test must be constructed so that the *true* case is the null one, as shown in Figure 2. Finally, if there is a null case and the other case consists of a single statement, then the logical IF may be used in the usual manner. A simple conditional counter is an example of this:

```
Pseudocode:  If ID is 2, then count.

Fortran:  IF(ID.EQ.2)KOUNT = KOUNT + 1
```

The *test* construct can be generalized for three-way selection--the arithmetic IF in Fortran. This is shown in Figure 3. Of course, any null case is simply omitted.

The *test* construct can be further generalized for *n*-way selection, using the computed GO TO or the assigned GO TO. (The assigned GO TO, whose pseudocode is essentially the same as that for the computed GO TO, will not be discussed here.) Figure 4 shows the *n*-way selection construct.

Students should generally be encouraged to use the two-way selection construct, since it corresponds to the *if-then-else* construct in other languages (such as COBOL, Pascal, Algol, and PL/1). They are shown how to nest *test* constructs corresponding to nested *if-then-else* groups. Students are expected to have no difficulty in transferring from the *test* construct to the *if-then-else* construct of other languages, but at this stage of our testing we cannot confirm this.

```
Test condition.
    False:  | Sequence for false case |

    True:   | Sequence for true case |
End test.
```

```
C --- TEST condition.
      IF(condition)GO TO s
C ---    FALSE.  Description of action.
                   | Sequence for false case |
           GO TO t
C ---    TRUE.  Description of action.
    s              | Sequence for true case |
    t CONTINUE
C --- END TEST
```

Figure 1. General two-way selection: pseudocode and Fortran code, where *s* and *t* are statement numbers.

```
Test condition.
    False:  | Sequence for false case |
End test.
```

```
C --- TEST condition.
      IF(condition)GO TO s
C ---    FALSE.  Description of action.
                   | Sequence for false case |
    s CONTINUE
C --- END TEST
```

Figure 2. Two-way selection with a null case: pseudocode and Fortran code, where *s* is a statement number.

```
Test condition.
    Negative:  | Sequence for negative case |

    Zero:      | Sequence for zero case |

    Positive:  | Sequence for positive case |
End test.
```

```
C --- TEST condition.
      IF(condition)s,t,u
C ---    NEGATIVE.  Description of action.
    s           | Sequence for negative case |
          GO TO v
C ---    ZERO.
    t           | Sequence for zero case |
          GO TO v
C ---    POSITIVE.
    u           | Sequence for positive case |
    v CONTINUE
C --- END TEST
```

Figure 3. Three-way selection: pseudocode and Fortran code, where *s*, *t*, *u*, and *v* are statement numbers.

```
Test condition.
    Case 1:  | Sequence for case 1 |

    Case 2:  | Sequence for case 2 |

             . . . . . . .

    Case n   | Sequence for case n |
End test.
```

```
C --- TEST condition.
      GO TO (s,t, . . . ,z),i
C ---    CASE 1.  Description of action.
    s          | Sequence for case 1 |
               GO TO r
C ---    CASE 2.  Description of action.
    t          | Sequence for case 2 |
               GO TO r
         . . . . . . . .
C ---    CASE n.  Description of action.
    z          | Sequence for case n |
    r CONTINUE
C --- END TEST
```

Figure 4. *N*-way selection: pseudocode and Fortran code, where *r*, *s*, *t*, . . . , and *z* are statement numbers and *i* is the index. (The assigned GO TO may also be used in the Fortran code.)

## The *Loop* Construct for Repetition

Repetition (looping) is the ability to repeat a set of statements. There are four types of loops to consider: test at the top (*loop while*), test at the bottom (*loop until*), test in the middle, and loop with a counter. The first three types can be grouped together in a generalized loop construct; the fourth is a special case of the general loop. The general loop construct is shown in Figure 5. (Again, the CONTINUE statement is used as the exit from the construct.) If *sequence 1* is null (contains no statements), then we have the case with the test at the top; if *sequence 2* is null, we have the case with the test at the bottom. (The *quit* statement is a loop control statement and is therefore not indented with the body of the loop, but aligned with the *loop* and *end loop* statements.)

The loop with a counter, the Fortran DO, is a special case of the general loop, having its test implicit in the *loop* (or DO) statement. This type of loop is shown in Figure 6.

The *loop* construct is easily transferable to the loop constructs of other languages, such as *do while* and *perform until*.

## The *Test* and *Loop* Constructs in BASIC

Because of the similarity of the logical IF in Fortran and the IF in BASIC, as well as the similarity of the DO in Fortran and the FOR in BASIC, the *test* and *loop* constructs in BASIC are analogous to those in Fortran. Figure 7 shows the two-way selection construct in BASIC. (Ageloff and Mojena say that the structure shown in Figure 7 demonstrates "poor design for the logic"--too many GO TO statements. (5)) Figure 8 shows the general loop construct and Figure 9 shows the loop with a counter.

## "GOTOless" Programming

The use of the *test* and *loop* constructs may actually increase the number of GO TO statements in a Fortran or BASIC program. The GO TO statements that are required for the flow of control through the constructs themselves, however, do not contribute to what is sometimes called "spaghetti logic." Rather they preserve the structure of the basic control units, selection and repetition, allowing one entry at the top of a unit and forcing one exit at the bottom. (After all, these unconditional transfers are explicit in the object code of *if-then-else* and *do while* constructs.) Students should be persuated that the GO TO statements used with *test* and *loop* are indeed intrinsic parts of the constructs and that GO TO statements should not be used for any other purpose, except when required for enhancing the simplicity and/or clarity of the program logic. Students should be forbidden, in any case, to transfer upward in a program, except, of course, from the bottom of a loop.

The comments that accompany the Fortran and BASIC implementations of the *test* and *loop* constructs, as well as indentation, are essential for identifying the constructs and specifying clearly the flow of control through them.

```
Loop.
    Sequence 1 in body of loop
Quit if condition.
    Sequence 2 in body of loop
End loop.
```

```
C --- LOOP.
    s     Sequence 1 in body of loop
C --- QUIT IF condition.
      IF(condition)GO TO t
          Sequence 2 in body of loop
        GO TO s
    t CONTINUE
C --- END LOOP
```

Figure 5. A general loop construct: pseudocode and Fortran code, where *s* and *t* are statement numbers. Either *sequence 1* or *sequence 2* may be null.

```
Loop for name = i to j by k.
    Body of loop
End loop.
```

```
C --- LOOP FOR name = i TO j BY k.
      DO s name=i,j,k
          Body of loop
    s CONTINUE
C --- END LOOP
```

Figure 6. General loop with counter: pseudocode and Fortran code, where *name* is the index of the loop and *i*, *j*, and *k* are the initial, test, and increment values, respectively.

```
Test condition.
    False:  Sequence for false case
    True:   Sequence for true case
End test.
```

```
100  REM--TEST condition.
110       IF condition THEN 220
120  REM--    FALSE. Description of action.
130              Sequence for false case
200            GO TO 310
210  REM--    TRUE. Description of action.
220              Sequence for true case
300  REM--END TEST
310            . . .
```

Figure 7. Two-way selection: pseudocode and BASIC code.

```
Loop.
    Sequence 1 in body of loop
Quit if condition.
    Sequence 2 in body of loop
End loop.
```

```
100  REM--LOOP
110            Sequence 1 in body of loop
200  REM--QUIT IF condition.
210       IF condition THEN 320
220            Sequence 2 in body of loop
300            GO TO 110
310  REM--END LOOP
320            . . .
```

Figure 8. General loop: pseudocode and BASIC code. Either *sequence 1* or *sequence 2* may be null.

```
Loop for name = i to j by k.
    Body of loop
End loop.
```

```
100 REM--LOOP FOR name = i TO j BY k.
110      FOR name = i TO j STEP k
120          Body of loop
200      NEXT name
210 REM--END LOOP
```

Figure 9. General loop with counter: pseudocode and BASIC code, where *name* is the index and *i*, *j*, and *k* are the initial, terminal, and incremental values, respectively.

## Conclusions

Unfortunately, at the time of writing, we have not completed the testing of this new approach and the new textbook. So far, however, the instructors are encouraged and the students seem to be responding well. The instructors report that they have had many fewer students go to them with programming problems than in previous semesters, which may provide some basis for optimism.

## References

1. Gary B. Shelly and Thomas J. Cashman. *Introduction to Computer Programming: Structured COBOL.* Anaheim Publishing Co., Fullerton, California (1977).
2. J. W. Perry Cole. *ANSI FORTRAN IV: A Structured Programming Approach.* William C. Brown Co., Dubuque, Iowa (1978).
3. T. E. Hull and D. D. F. Day. *An Introduction to Programming and Applications with FORTRAN.* Addison-Wesley Publishers, Reading, Massachusetts (1978).
4. Paul M. Chirlian. *Introduction to Structured Fortran.* Matrix Publishers, Inc., Portland, Oregon (1979).
5. Roy Ageloff and Richard Mojena. *Applied BASIC Programming.* Wadsworth Publishing Co., Belmont, California (1980), p. 112.

# PEOPLE MAKE THE DIFFERENCE

By Frank Barber
DeKalb County School System
3770 N. Decatur Road
Decatur, GA 30032

People make the major difference in educational computing. In the DeKalb County Instructional Computing Project, a major emphasis is placed on helping people (central office administrators, principals and teachers) to become computer literate. We believe that the preparation of students for the impact of technology on society cannot occur successfully until the adults in our system are prepared. This paper details the master plan for preparing our people for the major changes that are to come in the decade of the 80's.

## Introduction

The transformation of American society in the 20th century has been breathtaking! We have progressed from basically an agrarian society to a dynamic, industrial society with a higher standard of living and level of education than was ever thought possible. Our technical advancements stagger the imagination.

Hardware capabilities tantalize and tease the educator who can see the vast potential for computers. The development of good software for the miracle machines, even though it is painfully slow, adds to the excitement for those who can catch a vision of what could be. However, the most critical problems for educational computing are not in the world of hardware and software, but in the world of people. There are fears and misunderstandings about computers that raise questions about their presence and purpose in the classroom.

Will the computer suffer the same fate that other innovative applications of technology to education have experienced? The answer is yes, unless a plan for changing the skills and attitudes of people is thoughtfully devised, skillfully implemented, and carefully managed. The plan must have as the primary objective the development of computer literate adults. School people must know the capabilities of technology and how to use them to help teach and manage. There is a need for programs that will prepare teachers and administrators for the time when computers will be available to everyone.

## An Approach

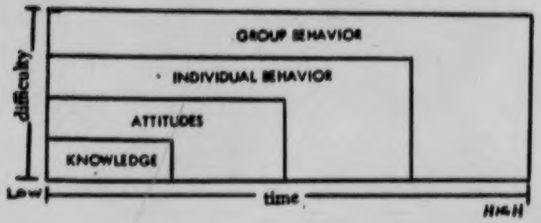
There are several methods for changing the behavior of large groups of people. All of them involve a time consuming, multi-faceted process.

In our method, there are at least four levels of change that are necessary if the computer is to become a tool used by all students, teachers and administrators for study, instruction and management. These are:

1) knowledge changes
2) attitude changes
3) individual behavior changes, and
4) group behavior changes

The most effective change process begins with level 1 and progresses through level 4.

In many cases, schools or departments are trying to begin on level 3 or 4. Inexpensive computers are being purchased without adequate instruction or support for the people who are expected to use them. The result is frustration, disappointment and reinforcement of the belief that technology has no place in the school house.



The chart shows the time and difficulty relationships involved in implementing change. Changes in knowledge and attitude are the quickest and easiest to make. Simply completing a well-designed course on the use of the computer in the classroom can produce remarkable results. However, a teacher may believe that using the computer is simple, appropriate, and will produce desirable results, but fails to employ new-found skills that were mastered in the course. Even with knowledge and attitude changes, when the door to the classroom is closed, traditional, comfortable methods are often still used. There are fears associated with trying something new that might not work. Generous applications of encouragement, patience, understanding and hand holding are often necessary before the individual behavior can be altered. Confidence in self and in the computer are essential ingredients for changes in what happens with young people on a day-to-day basis.

There will always be at least 10% of the teachers who will become trailblazers with innovative new methods. One must be careful about judging the success of an instructional computing program by the performance of one or two teachers. If the use of technology is confined to a limited number of teachers with a few students in a specific subject area, the broad goal of helping all students to become computer literate at a level commensurate with their general education will be lost. The initial experiences of the pioneers is very important. Their successes must be highly visible. The useful software and the dependable hardware must be widely publicized. Large numbers of teachers must be encouraged by the technology trend-setters to have hands-on contact with the computer. They must be able to observe how students respond when new methods are used. Then the difficult changes in large group behavior will begin to be realized and one will be able to see technology understood and used as an integral part of learning and living.

### An Application

The strategy for the development of computer literate adults in the DeKalb County School System embodies all four levels of change. It takes the form of a strong staff development program for teachers and administrators. The program is voluntary but employees are encouraged to participate through a variety of incentives. They can earn pay scale increment credit or state required certification renewal credit at no cost. They can earn graduate credit with reduced tuition, or best of all, they can earn a microcomputer for their school.

A computer is not placed in a school before at least one teacher or administrator has completed 50 class hours of instruction and another person who does not teach students all day, has completed a 10 hour course. The 50 hour course is designed to develop comfortable computer users, introduce the BASIC language, and to teach teachers ao apply new instructional techniques. The 10 hour course is the first 3 lessons from the 50 hour course. The courses begin to build the necessary knowledge and attitude base for the individual change. This approach makes the arrival of the microcomputer an exciting, highly desirable addition to the school.

Since completing course work does not mean that the computer will be used with students, before credit can be awarded, a follow-up visit is required. During the visit the teacher must demonstrate in a teaching situation, the use of skills and concepts mastered in the course. The observation period is not to be a one time show. The teacher must have used the computer with the students several times prior to the visit. This follow-up requirement provides the opportunity for the reward of seeing students respond with enthusiasm to the new tool. It goes a long way to establishing change in individual behavior.

As the first few teachers in a school share their success, and as other staff members want to know more about this new machine, provisions are made for faculty inservice sessions after school or on work days. In general, after this awareness worksession, teachers want to take at least the 10 hour course. During the 1980-81 school year, more than 200 teachers took the 10 hour course, more than 50 faculty inservice sessions were conducted and more than 150 teachers completed 50 hours of instruction.

A training effort of this magnitude could not be done by one person. During the summer of 1980, eight coordinators from the elementary instruction department completed the 50 hour course, the other coordinators completed 15 hours of instruction and the secretaries received a 10 hour course. The follow-up visits, the 10 hour courses and the faculty inservice sessions have been done by these coordinators. They have helped and encouraged the people in their schools. They have held the hands and solved the problems necessary to build the self confidence and computer confidence necessary to bring about changes in what happens on a day-to-day basis for students. The key to the success of the DeKalb School Computer Project has been thoughtful planning and competent, enthusiastic leadership given by the instructional coordinators.

Below is an outline for the 50 hour course.

Objectives:
You should be able to-

1) handle disks properly, boot the APPLE, and RUN programs
2) discuss current uses of the computer in instruction and the software that is available, and state several different problem solving strategies
3) make a diagram of the major computer components, explain in general how they work together to make a computer system
4) connect the components of a microcomputer or terminal and analyze and correct simple problems
5) write BASIC programs using DATA, READ, LET, PRINT and GOTO
6) write a BASIC program to accumulate a total and a program that will count
7) use the FOR and NEXT statements and the INT and RND function
8) use subscripted variables and the DIM statement, log on the Honeywell computer and RUN GIS
9) use nested loops and matrix variables
10) use GOSUB and RETURN and add music to a program

Conclusion

By 1985, the cost and capability curves for computers will level out at points where schools can readily afford hardware. Good software will be available. The challenge for educators is to have people ready. This can be done by planning and managing the change effectively. This must be done if education is going to prepare students for success in the real world.

# COMPUTER LITERACY AND COMPUTER USES
# IN EDUCATION: WHAT SHOULD THEY BE?

Fred Bell
4B01 Forbes Quad
University of Pittsburgh
Pittsburgh, PA 15260

During the 1970s a single computer programming course for secondary-school mathematics-education majors at the University of Pittsburgh evolved into a computer literacy program for teachers of any subject and any level from elementary schools to colleges. This program emphasizes three aspects of computer literacy — BASIC programming, computers and society, and computer uses in education. Teachers recognize the importance of computer literacy in their fields and we find that mathematics and science majors no longer seem to have affective or cognitive advantages in computer literacy courses. Rather, programming ability and motivation are related to general mental abilities and personality characteristics in some distinctive ways.

Reading, writing, and arithmetic taught to the tune of a hickory stick? Maybe at one time, but no longer. Now, in many schools, reading, writing, arithmetic, and other subjects as well are being taught to the tune of a microcomputer — the soft hiss of a cassette recorder or the rhythmic purr of a disk drive. Of course long before microtechnology resulted in reducing the cost of a computer to that of a television or a typewriter, some children were learning the three Rs at computer terminals — even before 1965. Now it appears that computers will not only be used in schools as teaching/learning aids for the three Rs but that computer literacy may take its place in school as a fourth basic literacy.

In 1980, the influential National Council of Teachers of Mathematics (NCTM) published its "Priorities in School Mathematics (PRISM)" for the 1980s, and one of its eight priorities was that "mathematics programs must take full advantage of the power of calculators and computers at all grade levels." The NCTM priorities were based upon a national opinion survey of nine categories of people: elementary school teachers, secondary school teachers, secondary school mathematics teachers, junior college and college mathematics teachers, mathematics supervisors, teacher educators, school principals, school board members, and parents as represented by PTA presidents. All of these groups strongly supported new curriculum materials and courses on computer literacy. Surprisingly, lay groups gave even stronger support for the use of computers in schools than professional educators. The PRISM study was a systematic attempt to assess people's preferences and priorities with respect to school curriculums. Opinion studies such as this, while not necessarily indicating which changes in school curriculums are best for society, are useful in predicting which curriculum changes may be readily adopted. So, it appears that there is considerable public support for teaching computer literacy in schools.

## COMPUTER LITERACY: WHAT SHOULD IT BE?

The number of answers to this question may be as large as the number of computer literacy courses that are being developed in schools, colleges, and elsewhere. However, there does seem to be a major disagreement about the extent to which a computer and learning a programming language should be incorporated into computer literacy courses. Courses called computer literacy range from lecture, discussion, reading, and film viewing courses about the impact of computers upon society to "hard-core" computer programming courses. The findings of this writer while teaching computer literacy courses and conducting research on computer literacy since 1968 indicate that computer literacy should be somewhere between these two extremes, which do exist in a real way.

A computer literacy program for school students developed at a school in Washington State was described in the "New Programs" column of the February 1980 Mathematics Teacher as requiring no experience with computers on the part of the teacher and no access to a computer in schools using the program. At the other extreme, a number of "computer literacy" courses in colleges turn out to be FORTRAN programming courses. The American Heritage Dictionary defines "literate" as: "1. Able to read and write." Extrapolating this definition to computer literacy, one might expect that a computer literate would be able to "read and write" computer programs in some computer language with some minimal competency. Consequently, a computer literacy course should prepare its taker to interpret computer programs written by others and to write his or her own programs. Some courses titled computer literacy require students to use programs written by others but offer no instruction in how to write programs. Such courses may develop in students an appreciation for computer applications and some understanding of the capabilities of computers, but they do not develop "computer literacy."

Semantics aside, the truly fascinating and intellectually stimulating aspects of computers are realized by people who have the satisfaction of creating their own programs which do precisely what they want done. As a TV-oriented society we tend to get our kicks watching other people do things rather than doing things ourselves. Numerous studies of teacher questions and test items in school classrooms demonstrate that most of what is taught in school is taught and learned at the lower end of the intellectual scale, at the knowledge and understanding levels. Little attention is given to the higher-level intellectual activities of analysis, synthesis, and evaluation — those cognitive abilities that we must use in order to write, debug, and improve a computer program. Computer literacy without programming is analagous to reading literacy without learning how to read. The reading illiterate can appreciate and understand the impact of reading upon society and the importance of reading, and even interpret the pictures in books and magazines. However, he or she is little more than an onlooker in the important human endeavor of reading. Two questions and their answers embody some of the essence of computers and computing. (1) Do you frequently fall asleep watching TV; why? (2) Do you frequently fall asleep working at a computer terminal; why?

If one accepts programming skills as a central aspect of computer literacy, then (1) what programming language should be learned; (2) which key words and commands should be emphasized; (3) what kind of programs should be written? First, BASIC is a popular language. It is standard on nearly all microcomputers and is used by the majority of schools that are using computers in classrooms. It is also a powerful language for important applications in education; it is easy to learn; and there is a large quantity of software (obviously not all of it is good) written in BASIC. However, there are other good languages that could replace BASIC in the future. Second, which key words and BASIC commands should be learned? The answer depends upon the type of people who are taking a course to become computer literate. That is, the answer depends upon the attitudes of the people not necessarily their ages. I have seen many 12 year olds who are good BASIC programmers. Even 7 year olds who can do a bit of printing and reading can write simple programs using RUN, LIST, SAVE and LOAD commands and PRINT, INPUT, GOTO, IF ... GOTO, STOP, END, and LET statements. Four year olds can RUN programs and correct their own typing errors while learning to read and spell three and four letter words or add and subtract single-digit numbers. So, the commands and keywords printed above might be minimal language requirements for computer literacy for young children. For 12 year olds and older people the sky is the limit. For example, for high school students and adults, the first three parts of A Guided Tour of Computer Programming in BASIC by Dwyer and Kaufmann together with a few special microcomputer programming techniques, including graphics, is an excellent start at the programming aspects of computer literacy. With respect to the kinds of programs that people should write, whatever is interesting or useful to the student is appropriate — within the bounds of good taste, of course. Students' desires to make a program do something special provide excellent motivation for teaching and learning additional programming skills.

Of course, computer literacy should include consideration of how computers are used in our society, how computers are changing our lives, and the social, economic, and personal aspects of a computer-based information society. The limitations of computer technology and how computers may be used in the future should also be considered. Other topics for a computer literacy course might be: computers and the law, computers and privacy, real and potential misuses of computers, and the science fiction aspects of computers. An entire body of literature including many new journals are also a part of the computers-in-society phenomenon and one should be aware of continuing sources of information about computers and computer applications.

## COMPUTERS IN EDUCATION COURSES

### COURSE CONTENT

Computer literacy as a University of Pittsburgh elective can be obtained by teachers and people preparing to be teachers through a three-credit graduate or undergraduate course, both of which are titled Computers in Education. The major goals for these two courses are the same, that is students will: (1) become acquainted with computer hardware, a programming language, computer terminology, and applications and limitations of computers in education and society in general; (2) learn to use a large time-sharing computer system and write, correct, and run programs on a time-sharing system; (3) develop a computer-enhanced curriculum module in their respective subject areas. Specific topics in these courses include:

1. Locating, selecting, and evaluating computer-based curriculum resources
2. Building a conceptual computer
3. Communicating with a computer
4. Learning and using BASIC
5. Computers in society
6. Microcomputers in schools

7. Computers in education
   - A. Administrative applications
   - B. Computer-science education
   - C. Computer-managed instruction
   - D. Computer-assisted instruction
   - E. Computer-enhanced problem solving
   - F. Computer based games and simulations
   - G. Computer-enhanced learning
   - H. Computers and motivation
8. Laboratory experiences and activities
   - A. Reviewing computer-enhanced curriculum modules
   - B. Hands-on use of computer-oriented instructional hardware and courseware
   - C. Selecting and evaluating courseware
   - D. Writing and improving a computer-enhanced curriculum module

These courses include lectures, discussions, a field trip to a computer-oriented learning center in a local school, reading assignments from magazines and journals, weekly programming assignments, quizzes, midterm and final examinations and a term project requiring each student to develop a comprehensive computer-enhanced curriculum module in his or her subject area. The majority of these modules approach, meet or exceed professional standards. In addition to a large selection of articles, the textbook used for learning BASIC is BASIC And the Personal Computer by Dwyer and Critchfield. The first two-thirds of this book is applicable to time-sharing systems as well as microcomputers. Students in these courses learn to use the following BASIC keywords with numeric and string variables and constants and appropriate programming "tricks."

DATA
DIM
END
FOR ..TO...STEP
GOSUB
GOTO
IF...GOTO...ELSE
IF...THEN...ELSE
INPUT
LET
ON...GOTO
ON...GOSUB
PRINT
PRINT TAB
READ
RANDOM
RESTORE
RETURN
NEXT
STOP

CONDUCTING A COURSE

Many, maybe even most students, begin a computer literacy/computers in education course with a fear of the unknown world of computers. This apprehension is reflected in many intelligent students being unable to remember and even follow simple instructions during their first terminal session. Some people don't even want to press a key without consulting the instructor. In fact, some students want to learn about computers but are afraid to take a beginning course and will not do so unless the course is required.

The first class session in the courses described above is devoted to overcoming these fears through free play and games with hand-held calculators; electronic games; an educational toy called "Alphie," that resembles a foot-high, music-playing, light-flashing, friendly robot that teaches pre-reading and pre-arithmetic concepts; a "Little Professor" that drills addition, subtraction, multiplication, and division at four levels of difficulty; and a microcomputer with a furry stuffed version of Sesame Street's "Grover" seated on top of the microcomputer. After RUN is typed, the microcomputer, with its previously loaded program, introduces itself as "a microcomputer named Shacky" (Radio Shack), and asks (on its screen) by name several students in the class to introduce themselves by typing their names, one at a time, on the keyboard. Appropriate personalized greetings are printed on the screen for each student brave enough to type his or her name. A few laughable nonsense programs are run, followed by some flashing light and cartoon graphics; then students gather around "Shacky" and take turns playing some hit-the-target games. All of these activities are accomplished with a lot of jokes by the instructor and laughter by students. By the end of class everyone has made the computer do something without harm to the person or the computer. About half of the students stay after class and play more computer-based games.

The next class session moves to a time-sharing classroom with a hard copy terminal for each student and video monitors in the four corners of the room for viewing the contents of the instructor's CRT. Students learn how to LOGIN and then RUN prepared programs to play math and word games. Finally, before LOGOUT, everyone types in a short "mystery program" from a handout and RUNs it to see what happens. Then everyone does a successful (after several attempts) LOGOUT, and computer anxiety is a thing of the past; although many frustrations come later in the course as people write, execute, and struggle to debug their own programs.

Each new BASIC keyword or pair of keywords is introduced by analyzing in class a RUN and LIST of a program containing familiar and new keywords to determine which lines of the program cause which lines of the output and why. Then as a homework assignment, each student writes, runs, and debugs a program that uses the new keyword or keywords to produce output whose format and content is specified by the instructor. The instructor and a graduate assistant have fequent, convenient office hours to assist students in debugging their programs, provided they bring a LIST of the program together with a sample bad RUN to the teacher's office. Many students prefer to come without a LIST or RUN, but soon learn that it is impossible to get help in debugging a program whose RUN and LIST are described orally by the student to the professor. The student's description of what happened "at the scene of the accident" (the RJE station) is seldom very accurate.

As the course progresses students help each other debug programs and eventially learn techniques and procedures for finding most of the bugs in programs by themselves. Near the end of the course most bugs are subtle ones such as 1 and I or O and 0 transpositions, nonprinting character errors, logical mistakes, or using a variable for conflicting purposes.

At times, new key words and programming techniques are motivated through the desirability of modifying and improving an existing program. For example, a drill-and-practice arithmetic program that prints RIGHT or WRONG following each answer can be improved by using IF...GOTO, ON...GOTO, IF...GOSUB, OR ON...GOSUB statements to select right or wrong answer subroutines, to vary right or wrong answer responses, and to diagnose probable reasons for certain wrong answers.

## COURSE EVALUATIONS

The following course evaluation form is used at the end of each course to collect students' opinions about the course. The rows of numerals under the rating scale (1 2 3 4 5) and beside each item are the frequencies of responses of each rating for each item. The numerals in the right-most column are respective mean ratings on a scale of 1 (poor) to 5 (excellent) for each of the 14 items. These data were gathered from the 19 people who completed the undergraduate version of this course in the Fall term 1980, which was the most recent offering of the course at this writing. It was noted that all of the poor and fair responses to any of the 14 items came from one student's form.

### COURSE EVALUATION FORM

Course: Computers in Education

Term: Fall, 1980

Please rate this course and the instructor on each of the following 14 items. Circle the code number that most closely represents your rating of the course and the instructor on each item. The numerical coding is: 1-poor rating, 2-fair rating, 3-average rating, 4-good rating, 5-excellent rating.

| | 1 | 2 | 3 | 4 | 5 | (Mean) |
|---|---|---|---|---|---|---|
| 1. Concepts were presented clearly and understandably. | 0 | 0 | 2 | 15 | 2 | 4.00 |
| 2. Examples and demonstrations clarified the concepts presented in the course. | 1 | 0 | 2 | 10 | 6 | 4.05 |
| 3. The instructor knew his subject. | 0 | 0 | 0 | 2 | 17 | 4.90 |
| 4. The instructor encouraged initiative and creativity on the part of the students. | 0 | 1 | 1 | 7 | 10 | 4.37 |
| 5. The course stimulated thinking. | 0 | 0 | 1 | 9 | 9 | 4.42 |
| 6. Content was presented in an organized manner. | 0 | 0 | 4 | 11 | 4 | 4.00 |
| 7. The course offered new viewpoints. | 0 | 1 | 5 | 8 | 5 | 3.90 |
| 8. The course contained worthwhile and informative material. | 0 | 1 | 2 | 11 | 5 | 4.05 |
| 9. Questions and opinions of members of the class were respected and discussed. | 1 | 0 | 3 | 10 | 5 | 3.95 |
| 10. Course content was presented so that you perceived its relevance to your interests. | 1 | 0 | 3 | 9 | 6 | 4.00 |
| 11. Ideas and materials were presented that will help you to become a more effective teacher. | 1 | 0 | 3 | 7 | 8 | 4.16 |
| 12. Assignments were relevant to you as a teacher. | 1 | 1 | 3 | 8 | 5 | 3.83 |
| 13. Your students will benefit from things that you learned in the course and assignments which you completed for the course. | 1 | 1 | 3 | 3 | 11 | 4.16 |
| 14. Your overall rating of the course. | 0 | 0 | 4 | 8 | 7 | 4.16 |

In addition to having students evaluate the course, I also wanted to assess their attitudes toward computers. The following scale was developed for this purpose. Items 1 through 9 assess each student's attitudes about his or her work with computers in the undergraduate course described previously. Items 10 and 11 determine sex biases with respect to learning about and using computers, and items 12 through 16 assess students' attitudes toward the importance of computers in society and people's understanding of them. These data were collected from 18 students who completed the undergraduate computer literacy course in the fall of 1979 and 19 people who completed the course in the fall of 1980.

The numerals to the right of the items and under the disagreement/agreement levels are frequencies of response. Responses were coded from -2 to +2 (strongly disagree to strongly agree) for items that are stated positively, with the codes being reversed for the negatively phrased statements in items 2, 8 and 10.

Please check the appropriate box. I am

MALE ☐

FEMALE ☐

| | STRONGLY DISAGREE | DISAGREE | UNDECIDED | AGREE | STRONGLY AGREE | MEAN |
|---|---|---|---|---|---|---|
| 1. I would like to learn more about computers | 0 | 3 | 6 | 15 | 13 | 1.03 |
| 2. Working with computers makes me tense or uneasy | 6 | 20 | 4 | 6 | 1 | 0.65 |
| 3. I would like very much to have my own computer | 2 | 3 | 7 | 18 | 7 | 0.68 |
| 4. I would like to take more computer courses | 2 | 3 | 10 | 15 | 7 | 0.60 |
| 5. I would like to use computers in some of my other courses | 3 | 3 | 10 | 14 | 7 | 0.51 |
| 6. I enjoy working with computers | 0 | 4 | 2 | 20 | 10 | 0.97 |
| 7. I feel confident about my ability to use computers | 0 | 5 | 9 | 18 | 5 | 0.62 |
| 8. I am not a person who works well with computers | 5 | 18 | 5 | 8 | 1 | 0.49 |
| 9. I can work with computers as well as most other college students | 0 | 6 | 15 | 11 | 5 | 0.41 |
| 10. Using computers is more for males than females | 22 | 12 | 3 | 0 | 0 | 1.50 |
| 11. Studying about computers is just as important for females as for males | 0 | 0 | 2 | 10 | 25 | 1.60 |
| 12. Every college student should have some understanding of computers | 1 | 2 | 1 | 18 | 15 | 1.19 |
| 13. Every college student should be able to write and run simple programs | 1 | 4 | 5 | 21 | 6 | 0.78 |
| 14. Every college student should learn about the role that computers play in society | 0 | 0 | 5 | 22 | 10 | 1.11 |
| 15. Computers can be useful instructional aids | 0 | 0 | 4 | 17 | 16 | 1.32 |
| 16. Computers are very important in our society | 0 | 0 | 4 | 20 | 13 | 1.24 |

## COMPUTER LITERACY AND STUDENT CHARACTERISTICS

In summary, based upon the data from the two questionnaires exhibited previously:

1. Students have a fairly high regard for the usefulness to teachers of computer literacy courses such as those described in this paper.
2. Neither males nor females expressed a sex bias toward the appropriateness of either sex learning about or using computers.
3. College students preparing to be teachers are fairly confident about their own abilities to use computers and have fairly positive attitudes toward computers after completing a single computer literacy course in college.
4. College students after taking a computer literacy course value the importance of computers in society and education; however they are not as confident about their own abilities to use computers as they are about the importance of computers in society.
5. It is possible to acquire a fairly good understanding of computer uses in society, learn a significant and useful number of programming skills, and be able to produce professional quality computer-enhanced curriculum materials through a single three-credit college course.

It is noteworthy to discover that mathematics majors (contrary to public misconceptions) do not appear to have any inherent advantages in computer literacy courses such as those described above, which include a significant amount of nontrivial computer programming. In spite of the fact that some of the non-math majors in these courses claim that math majors have an advantage, grades on the computer-enhance curriculum modules, programming assignments, and tests indicate otherwise.

Based upon discussions with and observations of students, personal characteristics that appear to influence success in learning computer programming in BASIC, and probably the other languages are:

1. An ability to think sequentially in small steps, i.e. algorithmic thinking ability
2. Being able to pay attention to details
3. Logical thinking ability
4. Tolerance for frustrations
5. Creativity
6. Above average intellectual ability
7. Perseverance and motivation

8. Independence
9. Organizational ability
10. Self confidence
11. Cognitive abilities of analysis, synthesis, and evaluation, including problem solving ability
12. Initiative, that is willingness to attempt difficult things

## BIBLIOGRAPHY

Bell, Fred. "Computer Literacy, School Mathematics and Problem Solving: Three for the Price of One." AEDS Journal 12 (Summer, 1979): 163-70.

Bell, Fred. "Classroom Computers: Beyond the 3 R's." Creative Computing 5 (September, 1979): 68-71.

Dwyer, Thomas and Critchfield, Margot. BASIC and the Personal Computer. Reading, Mass.: Addison-Wesley, 1978.

Dwyer, Thomas and Kaufman, Michael. A Guided Tour of Computer Programming in BASIC. Atlanta: Houghton Mifflin, 1973.

Johnson, David C. et. al. "Computer Literacy -- What Is It?" The Mathematics Teacher 73 (February, 1980): 91-96.

Johnson, Jerry. "Computer Literacy." The Mathematics Teacher 73 (February, 1980): 152-53.

Molnar, Andrew R. "The Next Great Crisis in American Education: Computer Literacy." THE Journal 5 (July/August 1978): 35-38.

National Council of Teachers of Mathematics. Recommendations for School Mathematics in the 1980s. Reston, VA: NCTM, 1980.

Suydam, Marilyn N. Mathematics Education. Alexandria, VA: Association for Supervision and Curriculum Development, February, 1981.

Sweitzer, Gary. An Interpretive Summary from the NCTM Project "Priorities in School Mathematics." Columbus, Ohio: ERIC Information Bulletin No. 1, 1980.

# ZERO BASE BUDGETING
# A MANAGEMENT SYSTEM APPROACH
# FOR EDUCATIONAL INSTITUTIONS

BY
ROBERT I. BERRY, Ed. D.
RESEARCH AND DEVELOPMENT CENTER
NORTHERN ARIZONA UNIVERSITY
FLAGSTAFF, ARIZONA 86011

ABSTRACT: This paper reviews in depth the Zero Base Budgeting Management system as it has been developed by Dr. Sam W. Bliss and myself for use by educational institutions. In essence, the ZBB management model as we have developed it is a management technique used to plan and allocate the financial resources of a school system to meet the needs of the community it serves, while at the same time providing many alternatives for the distribution of these resources. The ZBB management system model is a program planning and resource allocation system which requires administrators to justify their program and budget request in detail.

School boards and administrators are charged with the responsibility to plan, manage and control the financial resources placed at their disposal for operating educational institutions. Increasingly these school personnel have many pressing situations and issues to resolve in order to maintain and operate an educational institution that responds adequately to the educational needs of the students. One of the major problems facing school administrators is financial accountability. The public is demanding increased efficiency and effectiveness in the use of its tax dollars while at the same time placing constraints on the yearly increases in spending for education.

Traditionally most educational systems budget and account for these financial resources by object of expenditure classification. Items such as salaries, employee benefits, supplies and capital expenditures are linked to specific function areas such as instruction, support services and maintenance. This budgeting format while functional, does not relate the costs of specific planned expenditures input into the educational budget to the educational program. Hence, the educational managers, the community and administrators have little data with which to assess the results of educational activities or to decide among alternative methods for planning future program needs and expenditures.

In order to be more responsive to the public's demands for financial accountability school personnel must turn to a budgetary management system that will allow them to identify on a priority basis those programs deemed to be the most important to the school district and insure that full funding is accorded them. The budgetary management system must also identify those programs that may be logically and legitimately reduced or eliminated in order to better utilize their financial resources.

The Zero Base Budgeting Management System model provides the school district with the necessary information to make the best decisions possible. In essence, it is a management technique used to allocate the financial resources of a school system to meet the needs of the community it serves, while at the same time providing many alternatives for the distribution of these resources. The ZBB Management System model is a program planning and resource allocation system which requires administrators to justify their program and budget requests in detail. Emphasis is placed upon the evaluation of the planned effectiveness of the decision packages (programs) to the total educational system in relation to the districts' stated goals and obectives. Thus, the ZBB management system model is a planning technique that lends itself to an output-orientated system which provides the ability to analyze the cost versus the stated goals and objectives of the educational system.

The general benefits that can be realized through the use of the ZBB management system are:

1. Greater knowledge and understanding of programs that are functioning within the school district.
2. Increased level of communication both vertically and horizontally within the staff organization.
3. Identification of potential areas for resource savings, reorganization of programs, and new methods of program operation.
4. Development of additional opportunities to improve operations and productivity within the school district.
5. Improved systematic analysis of performance measures as related to resource costs, i.e., cost-effectiveness and analysis.
6. A budget that more clearly and directly supports the district's goals, objectives, and priorities.
7. A program budget for all programs and sub-programs, as well as an object-of-expenditure budget.
8. Program and fiscal accountability for the district.

## ZERO BASE BUDGETING AND THE TRADITIONAL BUDGET APPROACHES

Most traditional methods of educational budget preparation consists of requesting a proposed budget 'dream' list from each department or cost center. These budget request often are inflated with trade-off items and usually always represent only increased expenditures. Management controls this type of budgeting preparation by reducing these requests until the financial constraints of the district are satisfied.

Another traditional budgeting approach is the executive or cost-center budget system. This approach requires a varied and often complex formula usually based on student population to budget dollars for each school or department. Little concern is placed on how those resources are allocated or expended at the school level once the basic allocations are made.

The major problem with these two traditional budget preparation methods are that they represent a management system where by the information flows from the top management levels down to the lower management levels. Thus, these types of management systems are input systems only. These traditional budget preparation systems do not take into consideration the special and often unique needs of individual instructional programs across a school district. This problem is especially transparent when the school budget is construced utilizing a cost allocation formula. It must be assumed that each instructional or supporting program within the school district has the same funding requirements and is operating at the same efficiency level as other similar programs in the district. The ZBB Management System requires a flow of information and analysis from the bottom to the top. This procedure allows for the budgetary process to begin using input from the people who actually implement the programs; the classroom teacher or classified personnel at the work station. In this case increased personnel participation should result in greater support for the school districts' programs. It allows each operating program the flexibility to indicate to the school districts central administrators its' unique needs and requirements.

Another problem with the traditional approach of budget preparation is the lack of the necessary alternatives in terms of resources, program committments and program needs for the educational managers to make prudent decisions concerning resource allocation. The educational managers must have at their disposal all of the necessary components of decision making to make the best decision concerning resource allocation. The necessary components or alternatives required for decision making are:

1. continue funding at present level of effort
2. increase the funding level
3. decrease the funding level
4. do not fund the program at all (this option is usually not available to educational institutions)

The Zero Base Budgeting Management System not only requires all of the necessary components in financial terms, but also requires written narrative describing each alternative.

## ZERO BASE BUDGETING MANAGEMENT SYSTEM: A DEFINITION

The ZBB management system is a program planning and resource allocation system which requires each administrator to justify their program and budget request in detail. It requires all administrators to justify their program goals and objectives in finanical as well as descriptive terms. The ZBB management system requires that all programs consuming resources be described in "decision packages" which are systematically analyzed and ranked in order of importance to their decision units.

## ZERO BASE BUDGETING MANAGEMENT SYSTEM TERMINOLOGY

In the ZBB management system there are two unique terms that require some explanation these terms are:

1. Decision Units
2. Decision Packages

The decision unit may be identified as any budget or cost unit in the organization. Typically in a school district the decision units are generally defined in the organization chart. Decision units may be:

1. school sites
2. support service sites
3. instructional groupings
4. unit administrators

These decision units should be developed and identified at whatever organizational level that can be isolated and individually subjected to analysis and decision making, the organizational level where a meaningful decision can be made.

Within each decision is a collection of programs which consume resources. An example of programs within a decision unit defined as a school site would be

1. English
2. Math
3. Social Studies
4. Administration
5. Custodial
6. Pupil services
7. Athletics

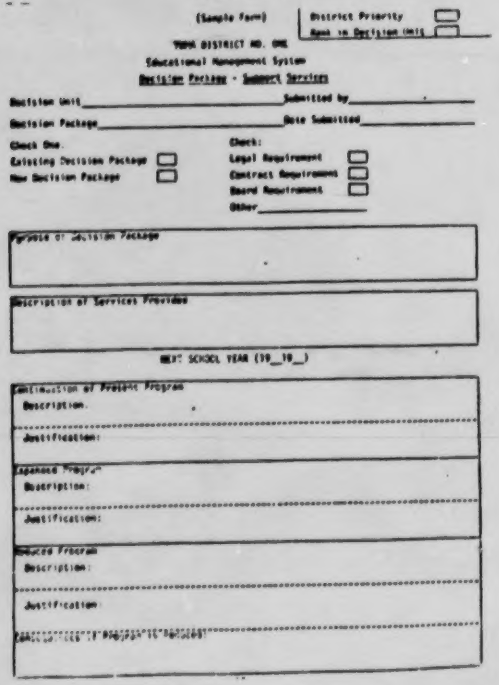
etc.

In ZBB terminology these programs are described in decision packages. Therefore a decision package is a brief, written description of instructional or support activities stating why they exist, what they do, what they plan to do, and alternative budget request for funding. The decision package allows management to:

1. Analyze and rank it in relationship with other programs that utilize organization resources.
2. Make a decision about acceptance, modification, or non-acceptance of the program.
3. Decide upon the funding level of accepted programs for inclusion in the budget.

Shown in figures 1 and 2 are examples of a decision package form as utilized by the Yuma Elementary Public School District in Yuma, AZ.

FIGURE 1



(Sample Form)

District Priority ☐
Rank in Decision Unit ☐

YUMA DISTRICT NO. ONE
Educational Management System
Decision Package - Support Services

Decision Unit ____________ Submitted by ____________
Decision Package ____________ Date Submitted ____________

Check One.
Existing Decision Package ☐
New Decision Package ☐

Check:
Legal Requirement ☐
Contract Requirement ☐
Board Requirement ☐
Other ____________

Purpose of Decision Package

Description of Services Provided

NEXT SCHOOL YEAR (19__19__)

Continuation of Present Program
Description:
Justification:

Expanded Program
Description:
Justification:

Reduced Program
Description:
Justification:

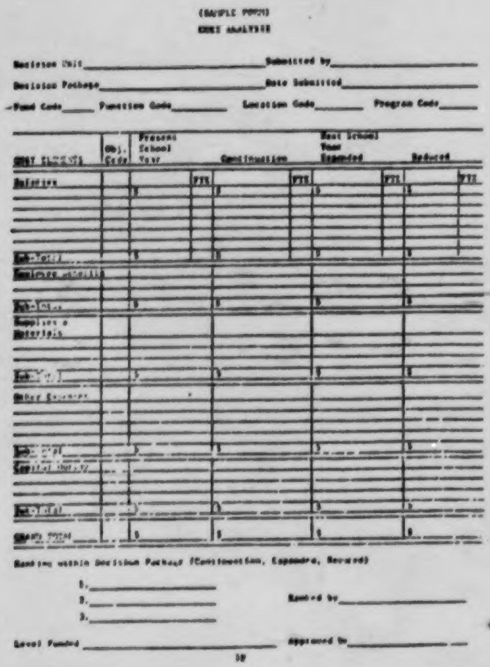
Consequences if Program is Deleted:

A decision package is written on the program or sub-program level and contains written justification, expectations, and from five to ten major objects of expenditure for each alternative educational plan. It is the combination of the decision packages (program plans) that creates the base and size of the total school budget.

Shown in figure 3 is a graphic representation of the decision package in relationship to the educational plan of the decision unit.

FILMED FROM
BEST COPY AVAILABLE

FIGURE 2



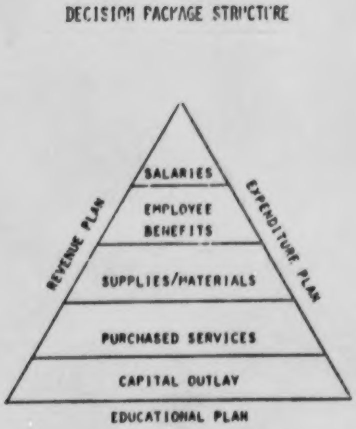
(SAMPLE FORM)
COST ANALYSIS

Decision Unit ____________ Submitted by ____________
Decision Package ____________ Date Submitted ____________
Fund Code _____ Function Code _____ Location Code _____ Program Code _____

| COST ELEMENTS | Obj. Code | Present School Year | FTE | Continuation | FTE | Next School Year Expanded | FTE | Reduced | FTE |
|---|---|---|---|---|---|---|---|---|---|
| Salaries | | $ | | $ | | $ | | $ | |
| Sub-Total | | $ | | $ | | $ | | $ | |
| Employee Benefits | | | | | | | | | |
| Sub-Total | | $ | | $ | | $ | | $ | |
| Supplies & Materials | | | | | | | | | |
| Sub-Total | | $ | | $ | | $ | | $ | |
| Other Expenses | | | | | | | | | |
| Sub-Total | | $ | | $ | | $ | | $ | |
| Capital Outlay | | | | | | | | | |
| Sub-Total | | $ | | $ | | $ | | $ | |
| GRAND TOTAL | | $ | | $ | | $ | | $ | |

Ranking within Decision Package (Continuation, Expanded, Reduced)
1. ____________
2. ____________
3. ____________
Ranked by ____________
Level Funded ____________ Approved by ____________

FIGURE 3



## ZERO BASE BUDGETING MANAGEMENT SYSTEMS OVERVIEW

The ZBB Management System has been designed to utilize three management planning levels and one budget report level. The first management level includes data entry, constructing of the master data file, selection of budget alternatives at the program level and the construction of school or location budgets based on selected alternatives. The second management level consists of utilizing the selected program alternatives to construct function budgets for prioritization within function areas. Principally the second management level consists of working with the selected program alternatives (level one) on a district basis. The third management level provides a third and final check of the new proposed budget. All of the three management levels are repetitive in nature and may be performed as many times as necessary. However, any actual dollar amount or other data changes associated with the data input from the decision packages or any change in the alternative ranking must be done with the level one management programs.

The reporting level consists of requesting the final budget reports. Currently there are four budget formats available: fund budgets, location budgets, program budgets and function budgets.

## ZBB - MANAGEMENT LEVEL ONE

The ZBB management system procedures at management level one were designed to provide level one decision makers with specific and summary type information about each of the decision packages input into the system. The activities performed during the first management level are:

1. Complete the decision package input from listing at least three alternative budget requests for each decision package (program). Usually these three budget requests are; a continuation of the present program plus inflation, an expanded program and a reduced or minimum program.
2. Prioritize the decision packages according to emphasis for the budget year for that decision unit. This prioritization is an important concept of the ZBB management model as it lends itself to better analysis of the decision packages.
3. Rank the budget alternatives of each decision package from first choice of funding to last choice of funding. This ranking technique lends itself to an excellent analysis method referred to as incremental analysis where by it is possible to start with the reduced or minimum funding level and incrementally add each budget alternatives to the minimum level, thus allowing an incremental analysis of what each budget alternative adds to the program described in the decision package.

Shown in figure 4 is the management system flow for level one. It presents the computer reports, activities and persons responsible for the process at this management level.

Shown in figures 5 and 6 are examples of the computer reports utilized by the level one managers in their analysis and display of their detailed decision package description. This report is utilized as an informational and prioritization process. Figure 6 represents the end result of the ranking and prioritization process, this report is used to report the activities and decisions made by the level one managers to the level two managers.

FILMED FROM
BEST COPY AVAILABLE

COMPUTER RESOURCE
EXAMPLE

ZERO-BASE BUDGETING
DECISION AREA RANK/PRIORITY REPORT
FOR LEVEL ONE

REPORT: ZBB-3C
LEVEL: 1
LOCATION: NORTH HOR

| DECISION AREA RANK | DECISION AREA NAME | FUNCTION | FIRST CHOICE | SECOND CHOICE | THIRD CHOICE |
|---|---|---|---|---|---|
| 1 | READING | INSTRUCTION | ALT-A | EXIST | ALT-B |
| 2 | MATHEMATICS | INSTRUCTION | ALT-A | EXIST | ALT-B |
| 3 | SCIENCE | INSTRUCTION | ALT-A | EXIST | ALT-B |
| 4 | SOCIAL STUDIES | INSTRUCTION | EXIST | ALT-A | ALT-B |
| 5 | HEALTH | INSTRUCTION | EXIST | ALT-A | ALT-B |
| 6 | LANGUAGE ARTS | INSTRUCTION | EXIST | ALT-A | ALT-B |
| 7 | SPEECH | INSTRUCTION | EXIST | ALT-B | ALT-A |
| 8 | PHYSICAL EDUCATION | INSTRUCTION | ALT-B | EXIST | ALT-A |
| 9 | ART | INSTRUCTION | ALT-B | EXIST | ALT-A |
| 10 | ADMINISTRATION | SUPPORT SERVICES | ALT-B | ALT-A | EXIST |
| 11 | AUXILIARY SERVICES | SUPPORT SERVICES | ALT-B | EXIST | ALT-A |
|  |  | TOTAL BUDGET REQUEST = | $435,635. | $430,600. | $400,200 |
|  |  | TOTAL F.T.E. = | 43.8 | 42.8 | 42.0 |
|  |  | DIFFERENCE FROM PRESENT = | $ 35,635. | $ 30,600. | $ 1,200. |
|  |  |  | 8.5% | 8.2% | .2% |

FIGURE 6

FIGURE 4

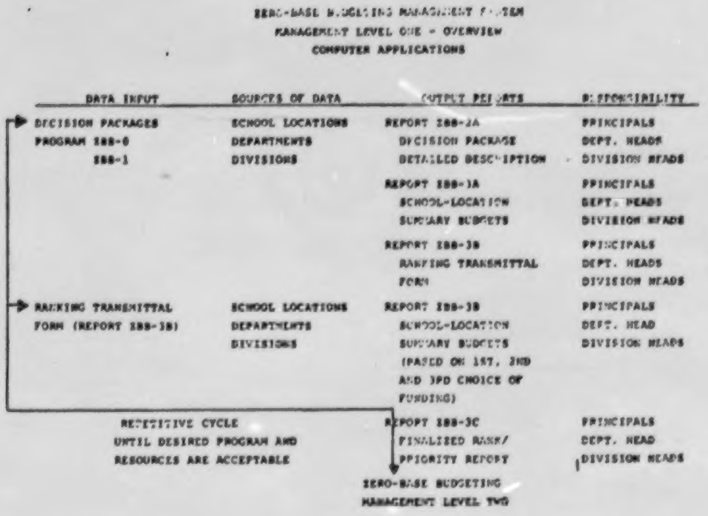

ZERO-BASE BUDGETING MANAGEMENT SYSTEM
MANAGEMENT LEVEL ONE - OVERVIEW
COMPUTER APPLICATIONS
DATA INPUT
SOURCES OF DATA
OUTPUT REPORTS
RESPONSIBILITY
DECISION PACKAGES
PROGRAM ZBB-0
ZBB-1
SCHOOL LOCATIONS
DEPARTMENTS
DIVISIONS
REPORT ZBB-2A
DECISION PACKAGE
DETAILED DESCRIPTION
PRINCIPALS
DEPT. HEADS
DIVISION HEADS
REPORT ZBB-3A
SCHOOL-LOCATION
SUMMARY BUDGETS
PRINCIPALS
DEPT. HEADS
DIVISION HEADS
REPORT ZBB-3B
RANKING TRANSMITTAL
FORM
PRINCIPALS
DEPT. HEADS
DIVISION HEADS
RANKING TRANSMITTAL
FORM (REPORT ZBB-3B)
SCHOOL LOCATIONS
DEPARTMENTS
DIVISIONS
REPORT ZBB-3B
SCHOOL-LOCATION
SUMMARY BUDGETS
(BASED ON 1ST, 2ND
AND 3RD CHOICE OF
FUNDING)
PRINCIPALS
DEPT. HEAD
DIVISION HEADS
REPETITIVE CYCLE
UNTIL DESIRED PROGRAM AND
RESOURCES ARE ACCEPTABLE
REPORT ZBB-3C
FINALIZED RANK/
PRIORITY REPORT
PRINCIPALS
DEPT. HEAD
DIVISION HEADS
ZERO-BASE BUDGETING
MANAGEMENT LEVEL TWO

FIGURE 5

Computer Example

ZERO BASE BUDGETING
DECISION PACKAGE DETAILED DESCRIPTION
TYPE - INSTRUCTION
FUNCTION-1100 SECONDARY INSTRUCTION

Report: ZBB-2A
Program-0200--English
Location-1150--
East Lake H.S.
Fund-0101--General Operating

| COST ELEMENTS | PRESENT FTE | FISCAL | NEXT FTE | FISCAL (Inflation) | ALTERNATIVE A FTE | (Increment) | ALTERNATIVE B FTE | (Reduction) |
|---|---|---|---|---|---|---|---|---|
| 1100-Cert. Teachers | 10.0 | $180,000 | 10.0 | $201,630 | 11.0 | $[illegible] | 9.0 | $[illegible] |
| 1300-Teacher Aide | 4.0 | $ 10,090 | 4.0 | $ 11,170 | 5.0 | $ 14,[illegible] | 5.0 | $ 14,[illegible] |
| Sub-Total-Salaries | 14.0 | $190,090 | 14.0 | $212,800 | 16.0 | $235,[illegible] | 14.0 | $195,4[illegible] |
| 2200-Medical Ins. |  | $ 10,000 |  | $ 12,000 |  | $ 13,1[illegible] |  | $ 11,[illegible] |
| 2201-Workmans Comp |  | $ 1,000 |  | $ 1,500 |  | $ 1,[illegible] |  | $ 1,[illegible] |
| Sub-Total-Employee Benefits |  | $ 11,000 |  | $ 13,500 |  | $ 14,[illegible] |  | $ 1[illegible] |
| 3100-Testing Service |  | $ [illegible],000 |  | $ 3,000 |  | $ 4,[illegible] |  | $ 3,[illegible] |
| 3200-Prof. Service |  | $ 1,500 |  | $ 1,500 |  | $ [illegible] |  | — |
| Sub-Total-Purchased Services |  | $ 4,500 |  | $ 4,500 |  | $ 6,[illegible] |  | $ 3,[illegible] |
| 4100-Supplies |  | $ 9,000 |  | $ 11,000 |  | $ 12,[illegible] |  | $ 7,[illegible] |
| 4200-Text Books |  | $ 12,000 |  | $ 15,000 |  | $ 24,[illegible] |  | $ 6,[illegible] |
| Sub-Total-Supplies/Materials |  | $ 21,000 |  | $ 26,000 |  | $ 36,[illegible] |  | $ 13,000 |
| 5200-Sound Equipment |  |  |  | $ 5,000 |  | $ 5,[illegible] |  |  |
| Sub-Total-Capital Outlay |  |  |  | $ 5,000 |  | $ 5,[illegible] |  |  |
| GRAND TOTALS | 14.0 | $226,500 | 14.0 | $261,800 | 16.0 | $29[illegible] | 14.0 | $22[illegible] |
| Difference/Present |  | -0- |  | $ 35,300 |  | $ 70,[illegible] |  | ($ [illegible]) |
| Percentage of Change |  | 0% |  | 15.60% |  | 31.[illegible]% |  | -1.01% |

Through the accumulation of all first choices of funding for all decision packages (programs) at a decision unit, it is possible to construct decision unit or location budgets based on the unique needs of the programs described on the decision package forms within the location. This represents a finite unit or building block approach to resource allocation whereby the management information flows from the bottom management levels to the top management levels. It is the sum of all the first choices of funding that determines the size of the decision units educational plan (the budget). Shown in figure 7 is a complete resource example of a decision unit budget based on a program.

FIGURE 7

SUMMARY SCHOOL BUDGET
FOR LEVEL ONE
FISCAL YEAR: 1979-80 BUDGET YEAR

Report: ZBB-3A
District: [illegible] County District #12
Location: East Lake H.S.

| BUDGET DESCRIPTION | PRESENT FISCAL | | | FIRST CHOICE | | | SUMMARY EFFECT | |
|---|---|---|---|---|---|---|---|---|
| Function: Secondary Instruction<br>Fund: General Operating<br>— Program: English | Amount | Cost Per Student Unit | Percent Budget | Requested Amount | Cost Per Student Unit | Percent Budget | Difference From Present | Percent Change |
| Cost Elements (Major) | $226,520 | $226.50 | 100.00% | $223,440 | $223.40 | 100.00 | [illegible] | [illegible] |
| 1-Salaries | $190,200 | $190.00 | 83.00% | $195,440 | $195.44 | 87.00% | $5,440 | [illegible] |
| 2-Employee Benefits | $ 11,000 | $ 11.00 | 4.00% | $ 12,000 | $ 12.00 | 5.00 | $1,000 | [illegible] |
| 3-Purchased Services | $ 4,500 | $ 4.50 | 1.00% | $ 3,000 | $ 3.00 | 1.00% | -$1,500 | [illegible] |
| 4-Supplies/Materials | $ 21,000 | $ 21.00 | 9.00% | $ 13,000 | $ 13.00 | 5.00% | -$8,000 | [illegible] |
| 5-Capital Outlay | -0- | | | -0- | | | | |

— The ZBB Computer Programs report major program totals. These major program totals may represent several first choices of funding for several decision packages.

## ZBB - MANAGEMENT LEVEL TWO

The Zero-Base Budgeting Management-Level Two were designed to provide the Level Two managers with specific and summary-type information about the decisions made at Level One, regarding the Level One managers' prioritization of programs and the requested allocation of resources. The Level Two Management cycle was designed primarily to make decisions on the allocation of resources at the Function-Major program level.

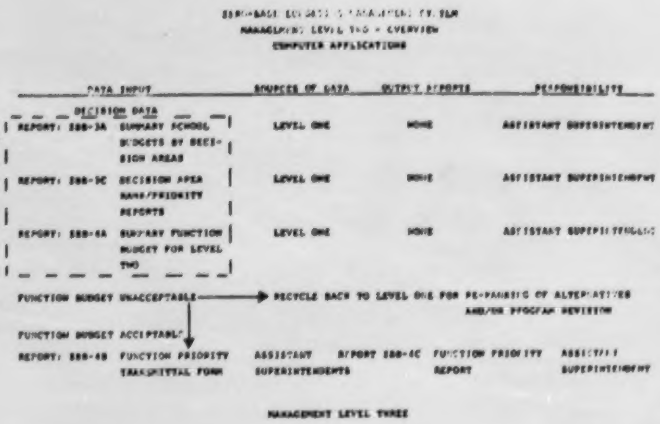
Sub-programs entered into the ZBB Management System as specific decision packages for a location (school or location) are collated to form function budgets (Instruction, Support Services) as major programs at Level Two. There are three activities performed at Level Two. They are:

1. To determine if the allocation of resources made-Level One (specific locations) are acceptable at the district level.
2. To provide guidance to the Level One managers for the re-ranking of alternatives and/or the allocation of resources made at Level One are not acceptable.
3. To prioritize all of the major programs within a specific function area, if the allocation of resources made at Level One are acceptable. This provides the Level Three managers with information regarding program priority within a function area.

In summary, the Level Two planning cycle consists of a check-and-balance type of activity to pull together specific location budget requests into acceptable district function budget request at the major program level.

Shown in figure 8 is an overview of the ZBB Management System for Level Two.

FILMED FROM
BEST COPY AVAILABLE



ZERO-BASE BUDGETING MANAGEMENT SYSTEM
MANAGEMENT LEVEL TWO - OVERVIEW
COMPUTER APPLICATIONS

| DATA INPUT | | SOURCES OF DATA | OUTPUT REPORTS | RESPONSIBILITY |
|---|---|---|---|---|
| DECISION DATA | | | | |
| REPORT: ZBB-3A | SUMMARY SCHOOL BUDGETS BY DECISION AREAS | LEVEL ONE | NONE | ASSISTANT SUPERINTENDENT |
| REPORT: ZBB-3C | DECISION AREA RANK/PRIORITY REPORTS | LEVEL ONE | NONE | ASSISTANT SUPERINTENDENT |
| REPORT: ZBB-4A | SUMMARY FUNCTION BUDGET FOR LEVEL TWO | LEVEL ONE | NONE | ASSISTANT SUPERINTENDENT |

FUNCTION BUDGET UNACCEPTABLE → RECYCLE BACK TO LEVEL ONE FOR RE-RANKING OF ALTERNATIVES AND/OR PROGRAM REVISION

FUNCTION BUDGET ACCEPTABLE

| | | | | | |
|---|---|---|---|---|---|
| REPORT: ZBB-4B | FUNCTION PRIORITY TRANSMITTAL FORM | ASSISTANT SUPERINTENDENTS | REPORT ZBB-4C | FUNCTION PRIORITY REPORT | ASSISTANT SUPERINTENDENT |

MANAGEMENT LEVEL THREE

FIGURE 8

Computer Resource Example

FIGURE 9
SUMMARY FUNCTION BUDGET
FOR LEVEL TWO FISCAL YEAR - 1979-80

Report: ZBB-4A
Major Function: Instruction
Function: Secondary Instruction

| BUDGET DESCRIPTION | PRESENT FISCAL | | | PROPOSED BUDGET | | | SUMMARY EFFECT | |
|---|---|---|---|---|---|---|---|---|
| Program: English<br>Location: East Lake H.S.<br>Fund: General Operating | Amount | Cost Per Student Unit | Percent Budget | Requested Amount | Cost Per Student Unit | Percent Budget | Difference From Present | Percent + or - |
| Cost Elements (Major) | | | | | | | | |
| 1-Salaries | | | | | | | | |
| 2-Employee Benfits | | | | | | | | |
| 3-Purchased Services | | | | | | | | |
| 4-Supplies/Materials | | | | | | | | |
| 5-Capital Outlay | | | | | | | | |

ONE LISTING FOR EACH LIFE PROGRAM
DISTRICT WIDE FOLLOWED BY A
PROGRAM SUMMARY.

Program: English
Location: Program Summary
Fund: General Operating

| | Amount | Cost Per Student Unit | Percent Budget | Requested Amount | Cost Per Student Unit | Percent Budget | Difference From Present | Percent + or - |
|---|---|---|---|---|---|---|---|---|
| Cost Elements (Major) | $700,000 | $175.00 | 100.00% | $537,000 | $153.43 | 100.00% | $163,000 | 23.30% |
| 1-Salaries | $140,000 | $ 35.00 | 20.00% | $102,000 | $ 29.14 | 19.00% | $ 38,000 | 27.10% |
| 2-Employees Benefits | $ 40,000 | $ 10.00 | 5.70% | $ 28,000 | $ 8.00 | 5.20% | $ 12,000 | 30.00% |
| 3-Purchased Services | $ 80,000 | $ 20.00 | 11.40% | $ 56,000 | $ 16.00 | 10.40% | $ 24,000 | 30.00% |
| 4-Supplies/Materials | $120,000 | $ 30.00 | 17.10% | $ 86,000 | $ 24.57 | 16.00% | $ 34,000 | 28.30% |
| 5-Capital Outlay | $320,000 | $ 80.00 | 45.70% | $265,000 | $ 75.71 | 49.30% | $ 55,000 | [illegible] |

Shown in figures 9 thru 11 are examples of the computer reports used in the management Level Two cycle. Figure 9 represents the function budget displayed by program within each decision unit and then summarizes all the specific programs into a district program based on the first choice of funding indicated for each program by the Level One managers. Figures 10 and 11 represent the prioritization of the major programs by function area. These reports are used as input documents to the Level Three managers.

FUNCTION PRIORITY REPORT FORM
FOR LEVEL TWO
FISCAL YEAR 1980

REPORT: ZBR-4C
DISTRICT: ADAMS COUNTY DIST.12
FUNCTION: SUPPORTING SERVICES

| | | | | | | CUMULATIVE TOTALS | | |
|---|---|---|---|---|---|---|---|---|
| RANK | MAJOR PROGRAM TITLE | FUNCTION | TOTAL F.T.E. | TOTAL AMOUNT | F.T.E. | TOTAL AMOUNT | PER CENT BUDGET |
| 1 | FINANCIAL SERVICES | SUPPORT SERVICES | 11.5 | $239,290. | 11.5 | $239,290. | 30.20% |
| 2 | MAINTENANCE | SUPPORT SERVICES | 20.0 | $465,720. | 31.5 | $705,010. | 89.00% |
| 3 | O/P & PERSONNEL | SUPPORT SERVICES | 1.0 | $ 87,300. | 32.5 | $792,310. | 100.00 |

NUMBER OF PROGRAMS = 3
TOTAL F.T.E. = 32.5
TOTAL BUDGET REQUEST = $792,310.
TOTAL PRESENT F.T.E. = 32.3
TOTAL PRESENT BUDGET = $735,580.
DIFFERENCE IN BUDGETS = $56,730.
% OF DIFFERENCE = 7.79%

FIGURE 10

FILMED FROM
BEST COPY AVAILABLE

FIGURE 11

FUNCTION PRIORITY REPORT FORM
FOR LEVEL TWO
FISCAL YEAR 1980

REPORT: ZBB-4C
DISTRICT: ADAMS COUNTY DIST. 12
FUNCTION: INSTRUCTION

| RANK | MAJOR PROGRAM TITLE | FUNCTION | TOTAL F.T.E. | TOTAL AMOUNT | CUMULATIVE TOTALS F.T.E. | CUMULATIVE TOTALS TOTAL AMOUNT | CUMULATIVE TOTALS PER CENT BUDGET |
|---|---|---|---|---|---|---|---|
| 1 | ENGLISH LANGUAGE ART | INSTRUCTION | 9.3 | $130,304. | 9.3 | $130,304. | 18.[illegible] |
| 2 | GEN SECONDARY EDUC. | INSTRUCTION | 15.6 | 290,721. | 24.9 | 421,025. | 59.[illegible] |
| 3 | MATHEMATICS | INSTRUCTION | 8.3 | 151,987. | 33.2 | 573,012. | 81.[illegible] |
| 4 | MEDIA SERVICES | INSTRUCTION | 8.8 | 131,323. | 42.0 | 704,335. | 100.[illegible] |

NUMBER OF PROGRAMS = 4

TOTAL F.T.E. = 42.0

TOTAL BUDGET REQUEST = $704,335.

TOTAL PRESENT F.T.E. = 41.0

TOTAL PRESENT BUDGET = $639,211.

DIFFERENCE IN BUDGETS = $65,124.

% OF DIFFERENCE = 10.20%

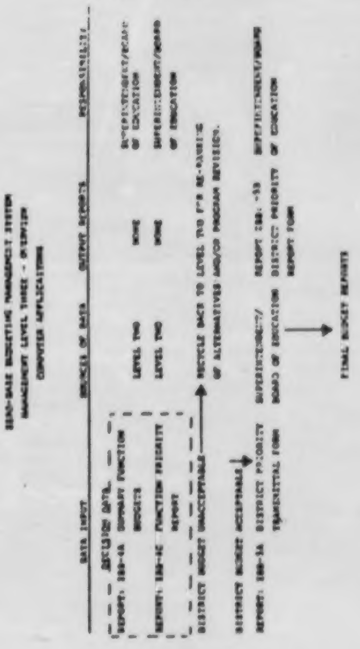
## ZBB - MANAGEMENT LEVEL THREE

The Zero-Base Budgeting Management System procedures at Management Level Three were designed to provide the Level Three managers with specific and summary-type information about the decisions made at Management Level Two. Utilizing reports from Level Two, the Level Three managers must review the major programs within the function areas and perform three activities. The three activities performed are:

1. To determine if the allocation of resources and prioritization of major programs submitted by the Level Two managers are acceptable at the district level.
2. To provide guidance to the Level Two managers for the re-ranking of alternatives and/or program revision that needs to be done if the proposed allocation of resources is unacceptable.
3. To prioritize all of the school districts' major programs if the proposed allocation of resources is acceptable.

In summary, the Management Level Three cycle consists of a check-and-balance type of activity to finalize a district major program budget and prioritize these major programs in order of importance to the needs and goals of the school district. Shown in figure 12 is the overview for management Level Three.

FIGURE 12


ZERO-BASE BUDGETING MANAGEMENT SYSTEM
MANAGEMENT LEVEL THREE - OVERVIEW
COMPUTER APPLICATIONS
DATA INPUT
SOURCES OF DATA
OUTPUT REPORTS
RESPONSIBILITY
DECISION DATA.
REPORT: ZBB-4A SUMMARY FUNCTION BUDGETS
LEVEL TWO
NONE
SUPERINTENDENT/BOARD OF EDUCATION
REPORT: ZBB-4C FUNCTION PRIORITY REPORT
LEVEL TWO
NONE
SUPERINTENDENT/BOARD OF EDUCATION
DISTRICT BUDGET UNACCEPTABLE
RECYCLE BACK TO LEVEL TWO FOR RE-RANKING OF ALTERNATIVES AND/OR PROGRAM REVISION.
DISTRICT BUDGET ACCEPTABLE
REPORT: ZBB-3A DISTRICT PRIORITY TRANSMITTAL FORM
SUPERINTENDENT/ BOARD OF EDUCATION
REPORT ZBB-3B DISTRICT PRIORITY REPORT FORM
SUPERINTENDENT/BOARD OF EDUCATION
FINAL BUDGET REPORTS

FILMED FROM
BEST COPY AVAILABLE

Figure 13 represents the prioritization process performed during management Level Three. This report represents a program priority budget for the district. It is very useful as a summary type of program budget for public review.

FILMED FROM
BEST COPY AVAILABLE

FIGURE 13

Computer Resource Example

DISTRICT PRIORITY REPORT
FOR LEVEL THREE
FISCAL YEAR: 1979-80

Report: ZBB-5B
District: Adams County No. 12

| | | | | | CUMULATIVE TOTALS | | |
|---|---|---|---|---|---|---|---|
| Priority | Major Program Title | Function Title | Total F.T.E. | Total Amount | F.T.E. | Total Amount | Per Cent Budget |
| 1 | English | Instruction | 62.0 | $ 920,000 | 62.0 | $ 920,000 | 12.96 |
| 2 | Mathematics | Instruction | 51.0 | $ 569,000 | 113.0 | $1,489,000 | 20.98 |
| 3 | Science | Instruction | 109.0 | $1,123,000 | 222.0 | $2,612,000 | 36.81 |
| 4 | Special Education | Support Services | 53.0 | $ 880,000 | 275.0 | $3,492,000 | 49.21 |
| 5 | Reading | Instruction | 56.0 | $ 768,000 | 331.0 | $4,260,000 | 60.03 |
| 6 | Maintenance | Support Services | 113.0 | $1,443,000 | 444.0 | $5,703,000 | 80.37 |
| 7 | Music | Instruction | 49.0 | $ 537,000 | 493.0 | $6,240,000 | 87.94 |
| 8 | Grounds/Buildings | Support Services | 59.0 | $ 856,000 | 552.0 | $7,096,000 | 100.00 |

Summary Totals

Number of Programs = 8
Total F.T.E. = 552.0
Total Budget Request = $7,096,000

ZBB - MANAGEMENT SYSTEM BUDGET REPORTING

The ZBB Management System prints from one to four district budget reports. All decision packages and their number one alternative figures are used to compute the budget figures. All reports show F.T.E. and dollar figures by detailed object codes and by major object codes.

Report ZBB-6A prints a function budget based on minor function codes, major function codes, and total district. For each break, four figures are printed: total staff F.T.E., total dollars, total major programs and average cost per student unit for all programs. At the end of the report, a summary table is printed showing each minor function code, title, total dollars, total staff, cost per student/unit and its percentage of the total budget.

Report ZBB-6C prints a location budget based on minor location code, major location code, and total district. For each break, four figures are printed: total staff F.T.E., total dollars, total major programs, average cost per student/unit for all programs. At the end of the budget, a summary table is printed listing all location codes, their titles, total dollars, total staff, cost per student/unit and its percentage of the total budget.

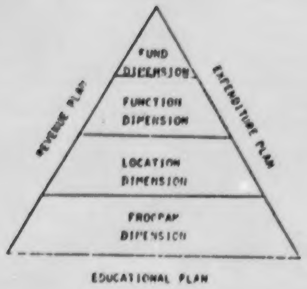
Report ZBB-6D prints a program budget based on major program code, minor location code, major location code, and total district. For each break, four figures are printed: total staff, total dollars, total major programs, and average cost per student/unit for all programs. At the end of the budget, a summary table is printed listing each major location code, title, total dollars, total staff, cost per student/unit and its percentage of the total budget.

IN SUMMARY

Zero Base Budgeting Management System is a planning method used to allocate the resources of a school district to meet the needs of the community it serves, while at the same time providing alternatives for the distribution of these resources. The computerization of the ZBB process allows for the consolidation of data collected from individual decision packages at Level One and utilizes this same data in many different forms through out the ZBB planning and report process.

The total school budget is like an equilateral triangle, the educational plan is the base of the triangle with expenditures and revenues as the other two legs. If any of the three side is changed, the other two sides must be changed. The Zero Base Budgeting process allows an orderly process for this change to occur by providing alternatives at the program or sub-program level.

FIGURE 14



TOTAL SCHOOL BUDGET

FUND DIMENSION
FUNCTION DIMENSION
LOCATION DIMENSION
PROGRAM DIMENSION
REVENUE PLAN
EXPENDITURE PLAN
EDUCATIONAL PLAN

FILMED FROM
BEST COPY AVAILABLE

# MUSIC EAR TRAINING AT OHIO STATE:
# THE PHOENIX/ORION PROJECT

Ann K. Blombach and Barbara A. Murphy
School of Music
The Ohio State University
Columbus, Ohio

In September, 1980, The Ohio State University School of Music began developing a series of computer-assisted lessons designed to meet the extremely varied needs of 350 music students enrolled in ear-training courses. The programs, written in the Phoenix computer language and run on OSU's Amdahl V7 computer, are intended to provide each student with individualized drill in all aspects of aural training. Magnavox Orion-60 terminals equipped with extensive graphic capabilities and touch sensitive screens will be interfaced with Rockwell AIM-65 sound synthesis microprocessors capable of playing up to four voices simultaneously. All programs will be competency-based, with the individual instructors determining the requirements for mastery. Extensive records kept for each student will be used for evaluation of the student's progress and diagnosis of the student's problems to provide maximum individualization of instruction.

## The Pedagogical Problem

Music ear-training has long been recognized as a necessary part of every musician's preparation, regardless of his field of specialization. All professional musicians--whether they be performers, conductors, teachers, theorists, or historians--are expected to be proficient in what is sometimes termed "aural musicianship"--those skills acquired through ear training. These skills are, in many ways, the musical equivalent of language skills. Musicians must be able to interpret and manipulate the elements of musical language, learning to perceive, understand, and express the relationship between music notation and musical sounds. Just as a child is not "born" knowing how to read and write his native language, so a musician is not born knowing how to read and write music. In some respects, the processes of learning natural language and musical language are very similar. As a child gains facility in using his native language, he practices reading the symbols of the language--e.g., in English, the series of letters, spaces, and punctuation marks that make up words and sentences. He is expected to understand the meaning of the written language whether he reads it aloud or silently. The child also learns to write the symbols himself so that he can express his own thoughts or write what other people say (a spelling test might be the closest analogue). Similarly, the professional musician learns 1) to understand the meaning of music notation--translating notation into sound, including "sight singing" as well as mentally hearing the written music without singing or playing it aloud; and 2) to use series of musical symbols to represent the musical sounds he hears--expressing sound in terms of standard music notation.

For the professional musician, ear-training or music language skills are a necessity. In fact, every accredited college-level music degree program requires each music student to take a minimum of two years of ear-training courses. Typically, these courses consist of two primary components: sight singing, in which the student uses his voice to demonstrate that he can accurately translate music notation into sound, and music dictation, in which the student uses the symbols of music notation to write what he hears. Success in aural skills depends on complex interactions among aural perception (recognition and identification of sounds), memory, music-reading, and notation skills. Although such skills have been part of musicians' basic training for centuries, relatively little is actually known about the exact relationship between music reading and dictation skills. Many ear-training teachers believe these two seemingly opposite processes

---

We would like to acknowledge the substantial support and assistance of: OSU University Systems Computer Center; OSU Office of Learning Resources, Computer-Based Education; OSU Office of Academic Affairs; OSU Graduate School; OSU School of Music; OSU College of the Arts; and Goal Systems Corporation.

are closely related since students often improve in one skill as they become more proficient in the other. And, as in learning other musical skills, practice seems to be essential to improvement.

Unfortunately, neither sight singing nor dictation is amenable to solitary and unaided study. Sight singing requires someone with a practiced ear to listen to the student and correct errors. Dictation requires someone to generate the sounds which the student is to notate, and then to check the accuracy of the student's notation. The most common method of teaching ear-training skills involves some combination of classroom work and taped drills available for practice outside of class. Neither approach is very effective, partly because beginning students demonstrate such a wide range of abilities--from those few who cannot even read music notation at all to those few who are extremely skilled in most aspects of aural musicianship. Because their expertise and their rates of improvement in the various skills depend not only on their differing innate abilities but also on their widely differing musical backgrounds and experiences, no two students have exactly the same problems. From an educational standpoint, each student's learning experiences (usually in the form of practice sessions) should ideally be tailored specifically to his special needs in order for that student to reach the highest level of success possible. In the traditional classroom setting, however, such individual attention is necessarily quite limited. Teachers must direct their attention to the "average" student with the "average" problems, even though classes consist of individual students with individual problems. The "average" student who happens to need practice on the particular skill at the particular level of difficulty offered in a given class session may benefit greatly, while the competency level of the remaining students in the class may improve very little.

A well-organized series of taped exercises can provide for more variation in students' music dictation abilities and needs than would be possible in classroom drills. Many colleges and universities, including Ohio State, have developed extensive libraries of taped materials which allow students to practice dictation skills outside of class so that teachers can concentrate more on developing sight-singing skills in classroom work. Ear-training books with accompanying taped exercises are also available for those schools which have not developed their own music dictation programs. Even with taped materials, however, students are restricted to a set number and sequence of exercises, and there is a practical limit to the variety and number of tapes available for practicing each aspect of music dictation skills. Students who practice the same taped exercises again and again--as they might whenever they encounter a particularly troublesome ear-training problem--soon unintentionally memorize the correct answers, thereby rendering the taped exercises useless. Usually the student himself is responsible for determining the correctness of his answers, often not until after he has completed a series of exercises. Lacking immediate reinforcement of correct answers and diagnostic feedback to help him correct his wrong answers, the student inevitably practices his own errors--a procedure hardly conducive to improving the student's ear-training skills. Without any mechanism for evaluation of errors, diagnosis of problems, or suggestions for improvement, taped exercises often prove both inefficient and ineffective. Students sometimes improve nonetheless, but many of them need something more than the ear-training tapes available to them. In their view, plodding through the taped examples, week after week, is an experience neither stimulating nor rewarding.

In short, musicians agree that music ear training is vital, but for many students, we are unable to provide classroom learning experiences or taped exercises adequate to satisfy their educational needs. As a result, students completing their required two years of ear-training study often are not as proficient in aural musicianship and music-reading skills as professional musicians should be.

## The Search for a Solution

One possible solution which has become very popular during the last several years is to provide the students with individualized computer-assisted instruction in music dictation. Ideally, the computer can serve almost as a private tutor for each student. The student's answers are evaluated immediately, with appropriate suggestions to the student whenever necessary. In the more sophisticated systems, the sequence and number of exercises are determined by the computer's diagnosis of the student's problems. Perhaps the two best known systems are the GUIDO ear-training lessons available through PLATO and the various new ear-training lessons being developed to run on APPLE computers. However, neither of these systems seemed appropriate for use at The Ohio State University. The APPLE capabilities--in terms of both data storage and graphic display--were inadequate for our needs. The big advantage was its relatively low cost, but we decided that a bargain system was not a wise investment if it could not satisfy our needs.

The GUIDO programs looked very promising at first, partly because two of our graduates (Fred Hofstetter and Michael Arenson) had organized the GUIDO lessons, based on Ohio State's taped music dictation exercises. Therefore, we expected little difficulty in fitting the GUIDO lessons into our existing curriculum. The graphic capabilities are excellent--the music notation looks like real music notation; we liked the idea that students could enter their answers by touching the screen instead of having to respond via a completely unmusical typewriter keyboard; and the sound generated by the "music box" is not too unpleasant. However, there were some major problems:

1) the costs of leasing terminals and hook-ups, not to mention the cost of telephone lines to the CDC computer in Minneapolis, were prohibitive. As one University administrator put it, "If we're going to spend that kind of money, we should be hiring people to work at Ohio State, not supporting a computer in Minneapolis."

2) the lack of local control over existing

programs. If we found some elements of the PLATO/GUIDO system inappropriate for our purposes, our only alternatives were to try to use them the way they were or to start over and develop our own courseware.

3) the lack of local control over data to be kept. At the time we were considering PLATO/GUIDO, there were no assurances that we would be able to keep or to access the kinds and amounts of data necessary for our research purposes. Because we are part of a major research-oriented university, one of our primary goals was to investigate, as thoroughly as possible, how different kinds of students learn to take music dictation. Without extremely detailed data for each student, such research studies would be impossible.

4) the lack of suitable mastery-learning (competency-based) courseware. We wanted more control of each student's progress than was possible through PLATO/GUIDO at that time.

For the University's administration, the overriding factor was the cost. Given that we already had a large computer and an excellent computer-based education authoring system on Ohio State's campus, we determined that the cost of access to PLATO/GUIDO over a five-year period would be approximately three and one-half times as much as buying our own terminals and developing our own courseware to run on OSU's computer. And, obviously, by following the latter course, we would have much more to show for our investment and our efforts by using our own system (e.g., we would own our terminals instead of merely having yet another year's "rent" due on PLATO terminals, and we would have our own courseware designed exactly according to our specifications). This year, therefore, we launched our pilot project. The University provided funds to hire Barbara A. Murphy as a programmer, to buy a computer terminal with PLATO-like capabilities, and to buy a music box that sounds musical.

## Phoenix/Orion Hardware

The hardware we are using to develop our ear-training programs consists of a Magnavox Orion-60 terminal and a Rockwell AIM-65 sound synthesis microprocessor.

The Orion terminal is equipped with a vector generator for graphic capabilities, a programmable character set of 96 characters, a plasma display, and a touch-sensitive screen. The Orion terminal provides high-resolution graphics, with a display screen consisting of a matrix of 512 x 512 addressable dot positions. The Orion is also capable of printing in four different modes: regular typescript (orange on a black screen), inverse mode (black type on an orange background), overstrike mode (printing over what is already there without erasing), and erase mode. Since the terminal has a plasma display, it will also support a rear-view slide projector, permitting simultaneous graphics display and slide projection. Input for the Orion may be entered either through the keyboard or by touching the screen at any one of its 256 touch positions.

Our Orion terminal is also interfaced to a music box, or sound synthesizer, being developed by Goal Systems Corporation of Columbus, Ohio. The music box, an AIM-65 microprocessor manufactured by Rockwell, is equipped with 32K RAM, 16K PROM, four I/O ports, and a PROM programmer. The box is capable of playing up to four simultaneous voices, each with a range of C0 (15 Hz) to C7 (2K Hz), a range approximately equal to that of a standard 88-key piano keyboard. For each of the four voices, the programmer can specify the attack, sustain, decay, and release parameters (the envelope) and the timbre, or sound quality. The timbre is controlled by definition of the waveform for the pitch, which can contain up to sixteen harmonic partials. The tempo and volume of the sound produced by the box can be controlled by the student. Some of the software for the music box was included with the purchase of the parts; modifications and enhancements of the basic programs are currently under way at Goal Systems. The music box will be controlled through the music ear-training programs. These programs are being written in the Phoenix computer language, also a product of Goal Systems. Phoenix, a greatly improved version of Coursewriter III, is an IBM-compatible language which Ohio State runs on an Amdahl V7 computer. We have found Phoenix to be a very good language for computer-based education purposes because:

1) it is a very efficient language. For instance, during Autumn Quarter 1980 at Ohio State, with an average of 55-60 students using the system at any one time, on the average day Phoenix used only 20-21 CPU minutes while handling 55,000 student responses. Response time is also minimal, averaging about 1.2 to 1.4 seconds during Autumn 1980.

2) it is a very effective computer-based education language. Phoenix has many string manipulative processes which allow for thorough checking of the students' answers. It is also capable of keeping extensive records of student performance through the use of auxiliary storage for maintaining individual data on each student and files for storing cumulative data on all students using the programs.

## Phoenix/Orion Software

The Phoenix/Orion ear-training lessons are being designed according to a mastery-learning format. Mastery learning is an instructional strategy that requires a student to learn, or "master," one unit of material before he can proceed to the next unit.[1] The student must demonstrate his knowledge of the material by scoring at or above a pre-determined cut-off point or competency level. Students who do not attain this score are required to continue working at that level until they are able to achieve competency.

---

[1] For more information about mastery learning, see John B. Carroll, "A Model of School Learning," Teachers College Record, 64 (May, 1963), 723-33; and Benjamin S. Bloom, "Mastery Learning," pp. 47-63 in Mastery Learning: Theory and Practice, edited by James Block (New York: Holt, Rinehart & Winston, 1971).

The software for the music ear-training programs consists of two main modes: the instructor mode and the student mode.

Each instructor may choose which lessons, or modules, he wants his students to work on each term. Modules available will include scales and modes, melodic and harmonic intervals, rhythmic patterns, tone groups (a combination of intervals and rhythmic patterns), melodies, two-voice counterpoint, triads, seventh chords, and harmonic dictation. The instructor may also set prerequisites for each of the modules he has chosen, along with the sections, or levels, of the modules he wants his students to work on. For each module there is a maximum of thirty levels plus a review section consisting of a mixture of exercises from those levels the instructor has indicated his students should complete. In addition, each instructor can set certain parameters for each level of each module. These parameters include the deadline date for each level, the number of times the student may have his answer evaluated, the elements of the answer he wishes the students to enter, and audio specifications such as the maximum number of times the student may hear the example, the speed of the presentation, and whether the student may have the music box play both his incorrect answer and the correct answer. The instructor may also set a minimum competency level required for mastery of each section (e.g., 18 out of the last 20 responses or 90% must be correct in order for the student to achieve mastery of that section).

In the student mode, the student chooses the module he wants to work on when he signs on to the program. A list of the levels the student is required to complete for that module is then displayed on the screen. To the right of this list are printed the deadline dates as set by the instructor. To the left of the list, an asterisk indicates each level the student has mastered, and an "L" indicates which, if any, of those levels were mastered after the deadline date. If the student is using the lesson for the first time, he is given information about how to enter his answers and how his answers will be evaluated. If the student has worked on the lesson before, he may ask to review these directions if he wishes. After this material is presented, the lesson itself begins.

If the student has chosen to work on a melodic dictation lesson, for instance, the computer first picks a melody at random from a bank, or file, of melodies containing as many as 250 melodies for each of thirty different levels (i.e., a total of 7500 possible melodies for the melodic dictation module).[2] The computer selects the melody within two seconds, and the screen is erased and readied for student input. After the student hears the melody (with the sound generated by the music box), he may begin to enter his answer by touching the boxes corresponding to pitch names or bar line, accidentals, octave registers, and duration values. The student may touch these boxes in any order and may change his mind as many times as he wants. Unlike some other computer-assisted ear-training lessons (e.g., PLATO/GUIDO), our lessons do not require the student to enter all parts of each note correctly before he may proceed to the next part of his answer. Instead, our lessons permit the student the freedom of entering partial note information; for example, he may enter just the pitch name and accidental, just the pitch name and octave register, or just the duration value. The student may also leave blank spaces for notes he does not know at all. Our lessons also let the student edit his answer by moving forward and backward, changing what he has previously entered, completely erasing what he has entered, or inserting notes or bar lines between two elements of his answer that he has already entered. The student may also play the melody again, and he may change the volume and/or tempo of the audio at any time during the process of editing his answer. No evaluation is made of his answer until the student indicates that he is finished by touching the "evaluate" box. His answer is then evaluated, feedback is given, and his score is calculated. The feedback includes messages telling the student which pitch names, registers, and/or duration values are wrong; if he missed one note, register, or duration but the relationships between succeeding notes are correct; and whether the contour of the melody is correct. If the student's score on the melody is not at least 90% correct, the screen will be erased and only the portions of the student's answer that were correct will be replotted. The boxes for entering the answer are drawn again; the student may then edit his answer and have it evaluated again. If the student's score was 90% correct or better but not exactly correct, he is shown the correct melody and is allowed to proceed to the next question.

Once the student achieves mastery on a particular level, he may choose 1) to keep practicing at the same level, 2) to move on to the next higher level which the instructor has indicated he should work on, 3) to select another module, or 4) to sign off. If he chooses to keep practicing at the same level, he may change the audio specifications and the elements of the answer he wants to enter and have evaluated if his instructor has indicated he may do so (e.g., he may choose to enter only solfege syllables). The student then indicates the number of exercises he wants to do before his progress is checked again. When the student chooses to work on another module or to sign off, he is first presented with a list of statistics for this session. These statistics include the number of problems he tried, the number that were correct the first time he had them evaluated, and the total time spent on the module, both for this session and across all sessions on this module.

The statistics the student sees, however, are only a small part of the extensive records kept on each student for each module of the program. Among the information retained is the total number of times the student has had his answer evaluated, the total number of times he had the example played, and the number of exercises attempted at each level of the module. For other modules, such as intervals and harmonic dictation, the computer constructs "tables of confusions" which facilitate the analysis of patterns in student errors. These tables permit the computer to formulate individualized practice sessions where each student will drill only on those portions of the lesson with which he is having trouble. The tables are also available to the instructors so they can see what their students are having problems with and drill on these aspects in the classroom. We plan to use the statistics we are keeping both for improving and further expanding the ear-training lessons on the computer and for improving classroom drill in aural skills. The record-keeping capabilities of our system will also provide extensive opportunities for research into the acquisition of aural skills.

---

[2] For other lessons such as intervals, the exercises themselves will be randomly generated (i.e., the computer will pick a starting note and an interval at random, and will then construct the problem).

At this time the ear-training programs in The Ohio State University's Phoenix/Orion Project are still under development. In Autumn Quarter 1981 approximately 200 freshman music students will begin using these programs. We plan to expand the drills for use by approximately 150 additional sophomore music students beginning in Autumn Quarter 1982. Looking toward the future, we anticipate that our system will provide an attractive alternative to other computer-based ear-training systems, particularly for those many institutions which already have access to IBM-compatible computers. The Phoenix/Orion system promises efficient, effective, sophisticated ear-training programs which utilize the capabilities of a large computer, while maintaining maximum instructor flexibility and local control of both programs and student data.

# EFFECTIVE INTEGRATION OF COMPUTER METHODS IN THE SCIENCE LABORATORY

Thomas Boe
Instructional Coordinator
Minnesota Educational Computing Consortium
2520 Broadway Drive
St. Paul, Minnesota 55113

ABSTRACT: Currently, many computer applications in the science classroom are effective and appropriate uses of the medium. Others are used to replace rather than supplement actual laboratory experience. The position is taken that computer applications should be developed which integrate laboratory experience with computer use. Development and testing of scientific models, laboratory problem solving and using the computer to collect, display and analyze data are among the mehtods of achieving this type of integrated laboratory experience.

Fifteen years ago the computer was making its first inroads into elementary and secondary education in the high school mathematics and science departments across the country. In many cases the computer is still considered the sole domain of these two disciplines. Recently, however this pattern has changed drastically. Good computer applications have proven their value in almost every academic discipline. During the same time period, the amount of computer software for science has increased substantially. The modes of use for the computer in the science classroom, however, have remained relatively static for over a decade. It is the goal of this paper to point out largely unexplored computer applications particularly suited to a modern science education.

There are at least two distinct areas in which the computer may be implemented realistically into the science curriculum. The first area involves courseware to support traditional modes of classroom instruction. This may reinforce previous concepts and skills or be used as an alternative medium to present new concepts. Such a general goal is not unique to science education. Neither are the computer implementations unique to science. The main tools for such an approach are drill and practice programs, educational games, and tutorials. Nothing in this type of implementation differentiates science education from similarly valid applications in reading history or mathematics.

The second area of computer use in the science classroom is in "real-mode" applications, modes similar to actual applications of computers in the scientific world. It is this second area of real world applications that makes computing in science education such an exciting prospect.

The use of computers to mirror real-world applications is becoming accepted by other educational disciplines, primarily business education. Business teachers have realized that accounting, payroll procedures and even typing are functions that are routinely performed in the business world today. The computer is a ubiquitous feature in the modern scientific world. Unfortunately, "real-mode" computer applications such as scientific problem solving, computerized experimental data gathering and analysis and realistic scientific modeling, have been largely ignored in science education. In order to make recommendations regarding this deficiency, it is worthwhile to describe some of these common uses of computers technology in the scientific world today and compare them with corresponding uses in the classroom.

## I. Problem Solving

All science consists of problem solving at some level. For the purpose of this paper, "problem solving" will be interpreted in a very narrow sense. A commonly used term for this activity in science is "number crunching". The computer's ability to process numerical data to reach a numerical result has been a great boon to scientists and engineers. Statistical analysis of large volumes of data, considered impossible forty years ago, is now routinely carried out in the scientific community. Some scientific problems, such as those involved in space flight, are not completely solvable with a finite number of calculations. The use of successive approximation methods and electronic computers now produce approximate results to such problems within any necesary degree of accuracy.

Problem solving is one of the "real-mode" computer applications that has been implemented to some extent in science classes since the beginning of classroom computing. Most of this work, however, has been done very informally with the initiative resting solely on the student. Before the advent of the hand calculator, for instance, students discovered that they could write simple programs to calculate the molarity of a solution in chemistry or calculate the maximum height of a projectile in physics.

The pedogogical value of simply using such problem solving programs in the science classroom is very limited. For class use such programs are only productive if the calculation being done is one which is repetitively used, as in an experiment situation with columns of data, and in which the solution process is so well known to the students that the repeated calculation is an unnecessary exercise. This describes a rare case indeed. This is not to say that computer problem solving is of no value. The program designers themselves can benefit greatly from the design process. The production of the program becomes a useful exercise in problem solving. In addition the students have realistically played the role of the scientist in directing the computer to solve a scientific problem.

### Recommendations

Computer problem solving in which students are involved in the actual design and programming should be actively encouraged. Problems may be assigned which are identified by the instructor as more easily solvable with a computer program than by hand calculator or other methods. Such computer applications problems should be compiled, organized by topic and distributed for teacher use along with sample solutions. Such an approach has found its way into secondary mathematics texts such as Houghton Mifflin's Algebra II and Trigonometry.[1] It is an equally valid tactic for the science curriculum.

Frequently those students most interested in the sciences are also involved in computer science. A healthy interdisciplinary approach to computer studies involving both science and mathematics teachers may be developed within a school. If an effort is made to coordinate such an approach, interdepartmental competition for computing resources and computing prestige should not develop. Introducing simple programming for the purpose of problem solving in the science classroom should boost interest in more formal computer science classes normally taught in the mathematics department.

## II. Experimental Data Gathering and Analysis

Many scientific experiments demand an ability to detect, record, and compile data faster and more accurately than is humanly practical. Specialized electronic gear such as multichannel analyzers, spectrographs, and chart recorders have been designed to perform these functions automatically. Computerization has added the ability to manipulate and analyze this data as well as detecting and recording it. Direct hookups between computers and such diverse detection instruments as radio-telescopes, radiation counters, and seismographic devices are common place in the scientific world.

In the area of science education, some commendable work on this type of instrumentation has been accomplished by Technical Education Research Centers (TERC), based in Cambridge, Massachusetts.[2] Using the low-cost KIM single board microcomputer, TERC has designed science experiments involving termperature measurement, frequency analysis, half life measurements, pulse height analysis and animal perception. In each case, the computer collects the data and provides an analysis or graphical display of the results.

The TERC development is of special interest because the KIM is just a 6502 microprocessor chip on a simple circuit board with only the bare essentials for input, output, and RAM memory. Since the 6502 is the heart of many popular microcomputers, such as the Apple II, Commodore PET, and ATARI, the TERC developed materials may be adapted for use with thousands of computers currently in schools across the country.

A few classroom laboratory applications for popular machines of this sort have already been developed including light and temperature measurement experiments for the Apple II [3] and physiology applications using the PET 2001 [4].

### Recommendations

TERC has demonstrated that a single microcomputer may be used to replace several expensive specialized laboratory instruments. In addition, the computer offers user controllable analysis and display capability not generally available with these specialized instruments.

Those institutions involved in educational courseware development should begin to produce software and accompanying experiment documentation designed to be used with the types of microcomputer systems already available in the classroom. Although a few pioneer applications of this type exist, more is needed with a higher level of program sophistication. This effort will produce better and more inexpensive laboratory capabilities for schools. In addition it will give the students a greater appreciation for the relationship between computer technology and modern experimental science.

## III. Scientific Modeling

The use of mathematical models to describe natural phenomena is an important process both in theoretical and applied sciences.

In applied science the mathematical basis for a model is accepted and a system is experimentally tested against this model. In planning the refueling of a nuclear reactor, various schemes for shuffling the fuel rods around are tried out in a computer simulation. The power output curves thus produced are then compared before a final arrangement is chosen. This same procedure is used for designing plant layouts, managing wild life, and optimizing chemical processes.

Frequently, models are also used after the fact. The operation of a device such as an airplane wing or an engine is tested in the laboratory then its performance is compared against its ideal operation as generated by a computer model. In any case, however, the modeling process is linked to an actual experimental trial.

In theoretical studies the roles of the model and the experimental data are reversed. The experimental results are known from a series of cases, whereas, there is no satisfactory mathematical description or explanation of the phenomenon. To resolve this problem, a series of mathematical models are developed and tested against the experimental data.

In science education the "simulation" type of program most closely resembles the process of scientific modeling. Many of these simulations are both instructive programs and good uses of computer capabilities. Conspicuous by their absence, however, are simulations that are tied to actual experimental study or even mirror accurately the computer program phase of a scientific modeling situation.

The typical simulation program for the science classroom is a surrogate experiment. This is justifiable in any case in which an actual experiment is impractical because it is too costly, dangerous or time consuming to be done in the laboratory. Satellite motion, long term population studies, and the operation of a nuclear power plant are all good candidates for such an approach. In many cases, however, such programs unnecessarily replace basic laboratory experiences. For instance, replacing a $15 burette with a $1500 microcomputer in order to do a titration experiment, creates an unrealistic situation which is not as learner-involved as the real thing. Unfortunately, several programs of this type are already available to the general public.

## Recommendations

Where possible new computer simulations should be developed to complement rather than replace laboratory experience. This conclusion is based on two considerations: First, science is a laboratory activity and any new development should not cause science education to further diverge from its experimental foundations. Secondly, the real-world uses of simulation in scientific modeing are all related to experimentation; tying a simulation to a related laboratory activity creates a situation that realistically mirrors scientific uses of the computer. This should result in a better understanding of modern science and its relationship to computing.

Several different approaches for producing integrated simulation - experiment modules are possible for courseware developers. Perhaps the most widely adaptable method is to take a good experiment that is widely used and provide a computer component. For example, a common physics laboratory on certripetal force involves manipulation of several variables including orbital radius, accelerating force, mass of the orbiting object, and the orbital period. If the data from each trial of the experiment is entered into the computer a best fit curve may be generated using regression analysis techniques. This may be compared against student curves of the same data. In addition a theoretical curve (model) may be generated simultaneously on the computer screen for comparison with the data curve. This allows students to evaluate the accuracy of their experimental results without "giving away" the mathematical formula which they are trying to discover. Furthermore, consistent inaccuracies in the model itself may be pointed out by comparing various students' results. Such inaccuracies may be attributed to frictional forces or the like which the simple model would ignore.

Another approach to develoment of this type of simulation-experiment material is to start with an existant simulation experiment and develop a laboratory component for it. This type of approach has been taken by the Minnesota Educational Computing Consortium (MECC) with the program DIFFUSION.[5] DIFFUSION calculates the diffusion rates of gasses chosen by the student relative to a reference gas which is also student specified. A laboratory activity was devised which allows the actual testing of these relative diffusion rates. The computer is used to check experimental data and to provide additional data for completion of a student data curve. Using the computer as a 'black box' in this manner still allows the students to discern the mathematical relationships on their own.

Many experiment possibilities exist which are accurate represenations of scientific modeling techniques. For instance, data may be recorded and displayed on the computer and possible mathematical formulae may be typed into the computer for simultaneous display. This would allow the students to discover the optimum mathematical expression by trial and error. Biological problems such as population curve studies may be particularly adaptable to such an approach since the relationships are complex and various portions of the curve may need to be analyzed separately.

## Summary

Computer applications involving drill and practice, tutorials, and simulations are as applicable to science education as they are to other disciplines. Other computer modes of specific relevance to science education, however, need to be developed. The main focus for such new development should be towards computer use that strengthens rather than replaces laboratory activity. Experimental data gathering and analysis by computer, the linkage of computer modeling to actual experimental activity and student program design for computer problem solving would be the basis for such a laboratory-oriented approach to computing in the science classroom. These new applications should demonstrate to the student many of the practical uses of computer technology in the scientific world.

## REFERENCES

1. Dolciani, Algebra II and Trigonometry, Houghton-Mifflin (1978).

2. Hands On!, Technical Education Research Centers (TERC), Spring, 1979, page 24.

3. Kraft, N., Elementary Science Experiments for the Apple, Classroom Computer News 1:1, 18-19 (1980).

4. Smith, I.C.H., The PET 2001 and CAL: Its Use in the Physiology Laboratory, Comput.& Educ. 3:361-365 (1979).

5. MECC, "Diffusion", Science Volume 2, 1981.

# EFFECTIVE FEEDBACK IN COMPUTER ASSISTED INSTRUCTION

Linda Borry Hausmann

Minnesota Educational
Computing Consortium (MECC)

2520 Broadway Drive
St. Paul, MN 55113

ABSTRACT: Feedback capability is the computer's strongest asset as a teaching tool. Compare a computer with other tools. Textbooks do a better job of presenting text. Video tapes and films do a better job of presenting vivid and detailed full scale images. Only the computer accepts a student's response and does something with that response to reinforce or enhance student learning. Therefore, courseware authors should take whatever time is needed to incorporate effective feedback into their lessons, and similarly, courseware consumers should examine the quality of feedback provided in programs to be used with students. This paper examines different types of feedback, the pros and cons of each, considers criteria for selecting appropriate types of feedback, and suggests guidelines for developing effective feedback.

## Introduction

The goal of most authors of computer assisted instruction (CAI) is is to produce a product that can carry on a dialogue with the student. The product should be capable of presenting material, asking questions, interpreting the student response regardless of its style or spelling, and then responding to the student in the same way a tutor would - acknowledging correct answers, providing additional help and hints for wrong answers, and redirecting the lesson if the student has insufficient background to complete the lesson or has mastered the material. This interest in dialogues is triggered by the computer's unique capability to provide feedback. Compare the computer to other teaching media. Textbooks are effective in presenting printed matter and accurate illustrations. Programmed texts go one step further by asking questions, providing answers, and suggesting a course of action. But a programmed text requires students to shuffle pages and evaluate their own progress.

A cassette tape gives a student access to high quality sound reproduction. Slide-tape productions, movies, and video tapes add high quality color video capabilities, but these media place the student in a passive role. All students are treated identically. There is no opportunity for students to let the media know that they are lost, bored, or have questions. Also, students cannot rearrange the order of presentation.

The computer's greatest strength is its capability to respond to students based on their responses. While its text presentation capabilities are inferior to those of a textbook, and its audio and video capabilities are inferior to those of slide-tapes, film-strips, video tapes and movies, it can provide feedback to the student. This capability should be used to its fullest extent in all CAI packages.

For purposes of discussion in this paper, feedback will be defined as "any response by the computer to input from the user."

## Types of feedback

Feedback falls into four categories: information, punishment, reward, and restructuring. Many types of feedback could be classified in more than one category. A smiling face, for example, could be both a reward and a source of information. In the sections that follow, key features of each type of feedback are discussed.

### Information

Information feedback tells students something about their answers or lets them know how they are progressing in the material. This type of feedback can play an integral role in concept acquisition. While learning a new concept, students need an opportunity to test their ideas. Receiving the simple feedback of "correct" or "wrong" after answering a question posed by the computer aids students in testing their ideas and thus in acquiring the concept. The immediateness of the feedback provided by the computer hastens the acquisition.

One step beyond a simple "correct" or "wrong" is to give the student additional information. This can be done by highlighting key concepts, providing the correct answer, or demonstrating the error in a specific response. In this way students not only know whether an idea is right or wrong, but why.

In simulations and games, a less direct type of information feedback is used. The computer does not print "wrong" or "correct", but rather shows students the results of a decision or answer. For example, in a lesson about food chains, one could just ask the student, "Would a wolf usually try to eat a rabbit?" and then process the response; but a more interesting approach would be to let the students take the role of the rabbit, present them with the situation of coming in contact with a wolf and let them select a course of action from a set of options such as "run, ignore it, attack and eat it, or chase it out of your territory." The students could make a decision and learn the results. If they decided they could eat the wolf and later found themselves as the wolf's lunch, they would learn that their original decision was wrong. This style of feedback encourages more independence on the part of the learner.

Another type of information feedback is one that tells students how they are doing. This can be as simple as providing a score at the end of a lesson. Expanding this idea further, the computer can inform students about their progress in the material and also let them know if they will be allowed to skip material or be given extra help.

### Punishment

Some authors use punishment as part of their feedback. Examples include comments like: "That was a stupid answer", "With a score like that, you'd better hit the books.", or "Wrong, turkey."

Feedback of this type is judgemental and should be avoided. Students who find the lesson difficult are feeling uncomfortable enough without having a machine make nasty comments about their intelligence or ability. Teachers are taught to be non-judgemental. Let's do the same in CAI.

### Rewards

The third type of feedback is the reward. Examples include: playing a song, providing a clever animation, or presenting a piece of a puzzle for each correct answer. The puzzle will be completed when the student makes enough correct answers. The reward is best used to give the student a brief rest or to spice up dull material. For example, at the end of a lesson a student appreciates a bit of relaxation, therefore, in addition to presenting a score, an author could provide an occasional reward. The student enjoys the break and is more ready to begin the next lesson.

A puzzle format can make a lesson more exciting. There are some things which can only be learned through repeated drill. Varying the format of the drill and occasionally using some clever rewards can make the lesson more enticing and encourage the student to persevere.

Rewards are fun to develop and implement, and as such, may be overused by authors. Possible pitfalls of rewards are distraction, boredom, and unproductive use of resources.

Rewards may divert student attention. While very clever rewards may engage a child's interest, they may be so interesting that the child concentrates more on the reward than on the material it is reinforcing. Research suggests that tangible rewards - and rewards could certainly be classified as such, can reduce student performance by diverting the student's attention from the learning task.

Rewards can cause boredom and be counterproductive. The first time children see a clever reward, they are delighted, but there is a point at which the reward has a neutral value, and in some cases a point at which it actually becomes a negative reinforcer. Picture a drill situation where each correct response causes a rabbit to slowly rise out of a magician's hat. A child begins to use the program. When she gets her first problem right, the rabbit emerges from the hat. The child smiles, perhaps laughs, or even squeals with excitement. A short while later, after several correct answers, the rabbit's appearance produces little if any response from the child. After 10 or 15 minutes and many correct answers, the child might be heard to mutter "Hurry up, rabbit, I want to try then next problem".

Rewards can take extensive time and resources to develop. Authors must ask themselves if this time might not be better spent in fine-tuning the pedagogical aspects of the package.

### Restructuring

A third type of feedback is restructuring the lesson to better meet the student's needs. Since the computer can monitor student performance, and keep records on the number of problems missed and the specific types of problems missed, the computer can selectively repeat problems, skip to more difficult material, or provide extra help. All this is possible if the author is willing to devote time to it.

Some packages present material with different levels of help for the student. For example, in a lesson dealing with the parts of speech, there are several ways an author could ask a student to identify the first noun in a sentence: The author could simply tell the student to identify the first noun in the sentence. The author could print the definition of a noun and then ask the student to find the first noun in the sentence. Or, the author could print the definition of a noun , print an example sentence in which all nouns are underlined and then ask the student to identify the first noun in a sentence. If a student is performing very well with a given level of help, subsequent material can be presented with less help. Similarly, if a student is having problems, the lesson can provide additional help, hints, or diagrams with the problems that follow.

Students who are having severe problems or are doing exceptionally well should be allowed to terminate the lesson if alternative restructuring is not available. The student who is getting less than 50% of the problems correct is having problems. If the lesson cannot move the student to easier material or give more help, then it should suggest that the student get help from his teacher. Being forced to continue in a lesson that is too difficult can be devastating to the student's self-concept and attitude toward the materials. Similarly, the student who has answered 20 of 20 questions correctly has probably mastered the material. Forcing him to answer more questions would only serve to bore him. Students who are doing extremely well should be moved ahead, given fewer helps, or allowed to terminate the program.

### Considerations in designing feedback

Four things to consider when designing feedback are the age or experience level of the user, the user's attitude toward the lesson, the number of times the average learner will use the material, and the amount of restructuring possible for the specific lesson. Feedback must be appropriate for the age of the user. The same feedback that delights a child could insult an adult. The reading level of the learner must also be considered. In addition, authors must ask themselves how much experience the average learner will already have had with microcomputers. Those who have used a variety of packages will expect to see graphics comparable to what they have seen before.

The learner's attitude toward the material is important. Think for a moment about your most exciting learning experience. What made it so exciting? In answering this question, many people describe an experience where they mastered something: learning to ski, learning to play the guitar, learning to program a microcomputer, or completing a set of difficult math problems. In each of these examples, the motivation for learning was intrinsic. The learner decided to learn for the sheer joy and challenge of learning and mastering something. The learner was not learning in order to receive an extrinsic reward such as a "good job" or a graphics show. How often do teachers deprive their students of the joy of learning by assuming that all learning is drudgery and therefore must be extrinsically rewarded? Most materials can be intrinsically rewarding if well designed. An author who strives for intrinsically motivating materials will usually produce a product that is better than one developed by an author who feels the students will not find the material interesting and who therefore spends a disproportionate amount of development time adding non-essential graphics and sound.

An author should also consider how the product will be used. Feedback for a drill program that is used daily should differ in design from feedback for a simulation used once in the course of the child's education. Feedback for frequently used programs should be brief and informative. In contrast, more elaborate feedback can enhance a program designed to be run only once.

Decide how much restructuring is possible. Some instructional materials lend themselves well to restructuring. Materials designed to teach skills can often be enhanced through elaborate restructuring. Consider the case of a program that teaches students to identify musical intervals by ear. A simple approach would be to present an interval at random, ask the student to identify the interval, let the student know if the answer is correct, and then present another problem at random.

A better system would flag the problems already presented and answered correctly so they wouldn't be repeated.

A more elaborate system could incorporate a counter for each type of problem. A student who hears a major third and identifies it as a perfect fifth has problems not only with major thirds, but with perfect fifths as well. With counters, both the major third and perfect fifth can be flagged to be repeated. Each time an interval is missed, the counters for both the interval played and the one with which it was confused are increased. When a student gets a problem correct, the counter for that interval is decreased by one. Before selecting a new problem, the counters are used to adjust the probability that the students receives the types of problems that are causing them the most difficulty.

In couclusion, feedback capability is the computer's greatest strength in CAI. Authors should take the time needed to analyze their products and provide the user with the best designed feedback possible. By the same token, consumers of educational software should take time to see how feedback is handled in products they plan to purchase. Where possible, experiment with the program. Try a variety of things. Take the role of a very gifted student, a slow student, an average student. Did the program react differently to the three students? A good tutor would have very different dialogues with the three individuals.

# RESULTS AND IMPLICATIONS OF SOME RECENT RESEARCH ON CAI IN MATHEMATICS

by

Ann Brebner, H. J. Hallworth, H. M. Woetowich,
S. N. Mah and H. C. Huang

The results of three formal studies concerned with adaptive computer assisted instruction for teaching mathematics at junior high school and high school level are reported.

In the first case, student performance was compared to that of students taught in the normal classroom setting. In the second experiment, adaptive CAI was measured against similar materials in booklet form. In the final study, student gain in achievement from the adaptive CAI was compared with that of students using CAI programs employing a more linear approach.

Common findings raise implications regarding the structure of such courseware, and also suggest avenues for further research.

## INTRODUCTION

At a time when microcomputers are coming into the schools in rapidly increasing numbers, when more teachers are becoming computer literate, and when they and other professionals are beginning to write computer assisted instruction courseware in increasing volume, it seems opportune to examine some of the techniques employed in such courseware and assess their effectiveness on both student achievement and attitude.

Three studies in secondary school level mathematics undertaken in schools in Calgary have addressed this problem. All three investigations were concerned with courseware which took account of individual student differences and difficulties, and which contained extensive remedial routines. All were implemented on a Digital Equipment Corporation PDP 11/70 computer system with the courseware written in the BASIC-PLUS language, which provides the author with comprehensive string handling and file manipulation facilities. The courseware was accessed by terminals in the schools using telephone lines running at 300 baud.

However, in each investigation a different area of the curriculum was covered, namely logarithms, concepts relating to factors and fractions, and the arithmetic of signed integers. Further, in each study a different comparison was used to evaluate the effectiveness of the CAI courseware. In the first case, student performance was compared to that of students taught in the normal classroom setting; in the second, adaptive CAI was measured against similar materials in booklet form; and in the third, gains in achievement by students using adaptive CAI were compared with gains by students using CAI programs employing a more linear approach (Ma, 1980; Huang, 1979; Woetowich, 1980).

## ADAPTIVE CAI AND CLASSROOM INSTRUCTION

The first experiment, covering the teaching of logarithms at the Grade 11 level, was the most comprehensive. The topic was divided into 22 modules and three programs were developed for each: a tutorial, a drill and practice, and a test. These modules were then arranged in a hierarchical structure and a student was not allowed to attempt a module unless mastery of the pre-requisites had been demonstrated. However, for some of the modules further prior skills were required, including concepts in decimal arithmetic and exponential functions. Hence, programs were developed for these topics, but were used only for remedial purposes.

An instructional management program controlled the CAI programs, developing a path for each individual student through the instructional materials based on past performance information.

The effectiveness of the CAI unit on logarithms was assessed in a formal experiment carried out in an urban high school. Sixty students randomly selected from the Grade 11 classes took part, with 30 in the experimental group and 30 in the control group. The latter group received conventional classroom instruction on the same topic from a teacher who had taught courses on mathematics teaching methods at university level and was widely recognized as highly competent. It was therefore assumed that the instruction given to the control group was above average.

As no suitable standardized achievement test on logarithms was available one was developed by the researcher. Sixty multiple choice items based on the school textbook Modern Intermediate Algebra were constructed and given to a group of 414 students. Based on their responses the test was reduced to 20 items with discrimination indices equal to or greater than 0.4. The reliability of the test was also statistically investigated and found satisfactory. This 20-item test was then used as a pre-test, post test and retention test in the experiment. A semantic differential type attitude test was also developed and administered to both groups immediately following the treatment.

The achievement data were analyzed by means of analysis of variance and co-variance, followed by t-tests and multiple classification analysis tables for the comparison of means. Both groups showed a significant improvement in achievement from the pre-test to the post test and a significant drop from the post test to the retention test, which was administered three weeks after the post test. However, there remained a significant gain in achievement from the pre-test to the retention test. While there were no significant differences between the two groups, an inspection of the means indicated a trend for the CAI group to gain more from pre-test to post test and lose less from post test to retention test.

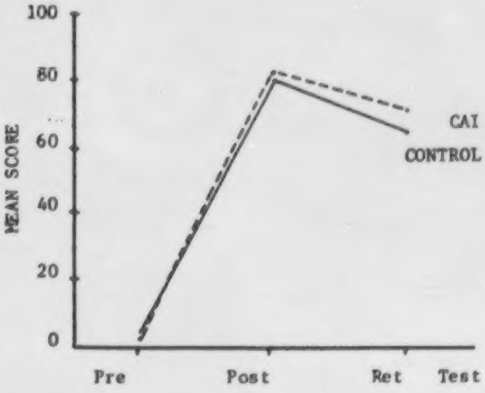


Figure 1.

Pre-test, Post Test and Retention Test Mean Scores

An analysis of covariance of the post test scores was conducted, with the pre-test scores and the time spent on the learning materials as covariates. The analysis showed both covariates to be significant. The CAI group achieved the same amount of learning as the conventional classroom instruction group, but in approximately half the time.

When the semantic differential data were subjected to discriminant analysis, it was found that the CAI group had a more positive attitude towards the learning of logarithmic functions.

Based on these data analyses, it was concluded that when used under the control of a teacher, the adaptive CAI system was as effective as a competent and experienced mathematics teacher, and that students using it reached the same knowledge level as their counterparts in the conventional classroom setting in significantly less time.

## LEARNER OR COMPUTER CONTROLLED ADAPTIVE CAI AND BOOKLET INSTRUCTION

In the second piece of research adaptive CAI was compared, not to the conventional classroom setting, but to individualized instruction given through the use of booklets. In addition, this investigation considered a second question, namely whether student control of routes and progress through the materials (the flexible condition) led to better achievement and attitude than computer or teacher control (the fixed condition).

Both the CAI programs and booklets covered six topics taught at the Grade 7 level. These were factors and multiples, prime and composite numbers, basic concepts of fractions, inequalities, addition and subtraction of fractions, and addition and subtraction of decimals. The topics were arranged in linear order and each was then broken down into five discrete units, a pre-test, instruction, drill and practice, a survey test and a post test. In the fixed mode students began by taking the pre-test of the first topic but from this point the adaptive features routed them on different paths according to their performance on the various tests. A good performance on the pre-test would automatically route the student to the post test; an average score would cause more test materials to be presented before a decision was made; and a poor result would invoke the instructional and drill sections again. Further similar branching occurred after each test. For example, after the post test was completed a student would either pass to the post test or pre-test of the next topic, or repeat the instructional sections and then attempt the post test again.

In the booklet method routing was similar except that the marking and decision making was done by the teacher.

In the flexible system, information about the units available was provided, and also suggestions were given on completion of a unit regarding the next appropriate one to attempt.

However, these could be disregarded and students could move through the materials in any way they chose.

A sample of 56 students was randomly selected to take part in the experiment. From this sample 14 were randomly assigned to each of four groups, CAI fixed, CAI flexible, booklet instruction fixed, and booklet instruction flexible. They were given a pre-test before working through the instructional materials, and a post test immediately upon their completion. These tests were specifically designed to test the students' understanding of the experimental materials. Further tests were also administered; these included a general mathematics achievement test, an intelligence test and an attitude measure.

A two-way analysis of variance with repeated measures was used to analyze the pre-test and post test data related to the main questions. This was used because it permitted the simultaneous investigation of both instructional methods and conditions. Results showed that a highly significant amount of learning took place and each of the four different combinations of methods and conditions was effective. However, there were no significant differences among the four groups in the amount of learning that took place.

An inspection of trends between group means suggested that there was a tendency for the CAI groups to learn more than the booklet instruction groups and for groups with computer or teacher control to learn more than those in which the students had control.

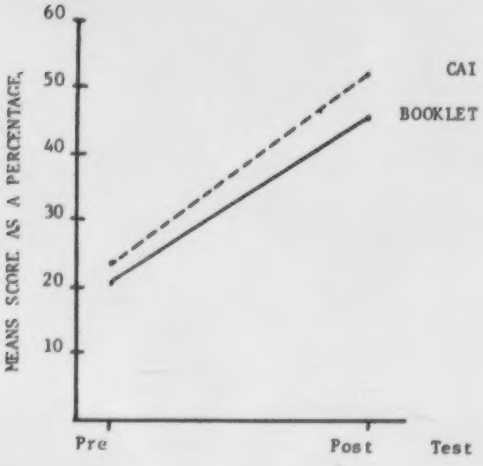


Figure 2.

Means of the Pre-test and Post Test by Method

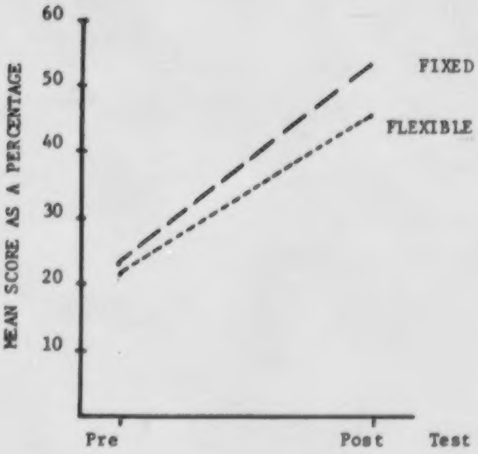


Figure 3.

Means of the Pre-Test and Post Test by Condition

Two-way analyses of co-variance were used to investigate the relationships first between the post test data and the mathematics achievement test, and then between the post test data and the intelligence test. It was found that the students' general mathematics achievement was significantly related to their performance on the post test, while the mental ability of this particular group of students was not.

Analysis of the attitude questionnaire showed that students in the CAI fixed group, the booklet instruction fixed group, and the CAI flexible group had a very positive attitude towards these treatments. However, the booklet instruction flexible group had a considerably less positive attitude towards the type of instruction they received.

## ADAPTIVE CAI AND LINEAR CAI

In the third study, the premise was accepted that computer assisted instruction is as effective a means of instruction as conventional classroom instruction or self-study booklets, and an investigation was undertaken to determine the relative values of various adaptive techniques which can be built into CAI programs.

The topic chosen was integer arithmetic, taught at the Grade 7 and 8 levels. The adaptive courseware consisted of eleven tutorial modules, designed to provide all the instruction required for the topic, and seven programs which gave drill and practice in addition, subtraction, multiplication, and division of signed numbers.

Two entirely different teaching approaches were employed. The first was the more mathematical method which used, for example, the concept of taking trips up and down a number line in addition and subtraction, and the perception of patterns of results in multiplication and division. The second approach to the topic was more verbal and practical in nature and was based on the postman stories developed by Davis (1967) in the Madison Project. The latter courseware was automatically presented when criterion was not reached on the other teaching materials. Other adaptive techniques included remediation sections for specific errors, such as adding absolute values when the signs are different, and the use of Wald Sequential Probability Ratio Test to determine mastery in some instances (Olivier, 1972).

To test the adaptive courseware a parallel set of materials was developed which covered the same topic with tutorial and drill and practice programs, but without any of the adaptive or remedial features.

A controlled experiment was carried out which involved a sample of 23 Grade 7 and 22 Grade 8 students. These were randomly assigned to one of the two treatment groups. One group received the adaptive courseware, and the other the parallel set of programs. All subjects received the same pre-test and post test, which was developed by the experimenter. The experiment was conducted over a period of approximately five weeks, with two terminals in a small junior high school. Statistical analysis of the data revealed that both groups improved significantly from pre-test to post test. There was, however, no significant difference between the mean scores obtained on the post test by the two groups.

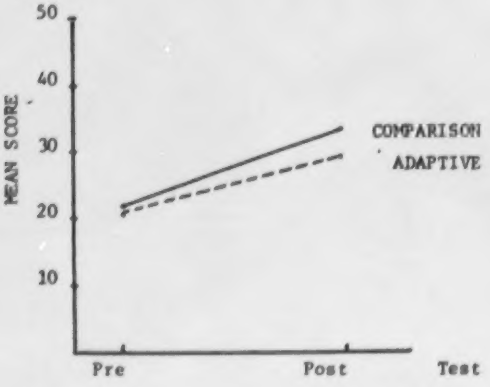


Figure 4.

Pre-test and Post Test Mean Scores

Further analysis of subsets of the data revealed no trends favouring one particular treatment over the other. It did, however, disclose one interesting point, namely that the remedial tutorial sequences were not used except in the concept of subtraction. Even analyzing this separately gave no indication that the alternative teaching strategy was more effective than simply repeating the program for which criterion was not reached. On the other hand, lack of use could not explain the apparent ineffectiveness of the remediation given in the drill and practice sections: this remediation was used quite frequently. However, it appeared that a short message to try again had an equal impact on the correction of student errors.

## CONCLUSIONS

The conclusions from the first study are similar to those reached in many other CAI experiments (Thomas, 1979). A tutorial system which is adaptive and allows individual paths and remediation for each student is as effective a means of instruction as the presentation of the material by a competent teacher in a conventional classroom setting, and students reach the same level of knowledge in considerably less time. In this particular case the average time spent by the CAI group was approximately half that of the other group. There may, however, have been a novelty or Hawthorne effect in the experiment, as none of the students who took part were normally exposed to the CAI form of learning.

Although there were no statistically significant results in the second piece of research, various trends were apparent. First, it appears that learner controlled instruction is inferior to computer controlled or instructor controlled sequencing of units. Achievement results are better if the decisions relating to the optimum paths to take through the materials are not made by the students. An author who is an experienced teacher knows the best approach to presenting the topic, while the students often feel apprehensive about having to make such decisions. This appears to stem, quite naturally, from the requirement of a decision based on little knowledge; in both CAI and individual booklet approaches to learning, much is hidden from the student. These forms of learning are very different from leafing through a text book, seeing the entire content and selecting a chapter to study.

The preference for computer controlled sequencing is not confined to school level students. A similar situation occurred in an experiment on the teaching of the statistic chi-square to undergraduate students (Taerum, 1974). The outcomes of this latter research, indeed, were somewhat confounded by the fact that students in the learner controlled group asked those in the computer controlled group what sequence they were following, and then chose the identical route. It appears that since Grubb's rather startling results in the late 1960's (Grubb, 1969), little evidence has been forthcoming to support learner control.

Another trend which emerged from the second study was that the CAI system was slightly more effective than the individual booklets in teaching the selected topics at the Grade 7 level. This could be attributed to the immediate feedback and record keeping facilities, especially routing, unique to CAI presentation. A number of factors may have contributed to this and militated against a significantly better performance by the CAI group, such as insufficiently concentrated exposure to CAI. Nevertheless, the fact that both methods were effective in teaching the topics indicates an important factor was that the materials were broken down into small logical steps for presentation to the students. This appeared to be the overriding criterion for the success of this experiment, dwarfing any effects from the method of delivery or amount of student control.

This contention is supported by the outcomes of the third study. In the tutorial programs the teaching steps were small and clear, and the sequencing was such that few students needed to access the second alternative teaching approach. Even in the cases where remediation was necessary, repetition of the original programs was as effective as the different explanation of the topic. Again, in the drill and practice sections, steps were also sufficiently small that knowledge of whether an answer was correct or not had an equal effect on students' learning as more elaborate remedial sequences which explained exactly why the answer was wrong.

In general, it can be postulated that the CAI method of delivery forces the courseware author to break the materials down into small sequential steps, and that if this is done well, there is little more a student needs except immediate feedback and a routing system which guides each individual through the materials according to performance on previous units. Too much branching and remediation can become confusing to the student and as a result can have no more effect than a totally linear approach (Wiens, 1975).

CAI is still in its infancy; hardware developments have outstripped the development of good courseware for many years, and the gap is ever widening. For authors, programming courseware is still novel and there is sometimes a tendency for them to be carried away by the facilities provided by the hardware, producing irrelevant graphics and gimmicks and then building some course material around them. Their students are often lacking in appreciation of their efforts. In several small studies students have often requested the deletion of a "happy face" or other spurious graphics because they feel the time spent looking at it is wasted; they prefer to be getting more information, or more drill and feedback on their answers.

Today one of the greatest needs in CAI research is concerned with what constitutes good courseware. The three studies reported here have attempted to answer some of the questions in three specific areas of secondary level mathematics. However, very much more needs to be done before CAI reaches its full potential as a powerful teaching technique.

## REFERENCES

Davis, R. B. Explorations in Mathematics. Don Mills, Ontario: Addison-Wesley, 1967.

Grubb, R. E. Learner Controlled Statistics. In R. C. Atkinson and H. A. Wilson (Eds.), Computer Assisted Instruction: A Book of Readings, New York: Academic Press, 1969, 167-172.

Huang, H. C. A Study of Computer-Assisted Instruction in Junior High School Mathematics: An Individually Self-Progressive System. Unpublished Doctoral Dissertation, University of Calgary, 1979.

Ma, S. N. A Study of Students' Learning of Logarithmic Functions Using an Adaptive Computer Assisted Instruction System. Unpublished Doctoral Dissertation, University of Calgary, 1980.

Nichols, E., Heimer, R. and Garland, E. Modern Intermediate Algebra. New York: Holt, Rinehart and Winston, 1969.

Olivier, W. P. Computer-Assisted Mathematics for Upgrading College Student Skills. Proceedings of the Canadian Symposium on Instructional Technology, 1972, 206-220.

Taerum, T. V. Computer-Assisted Learning in Statistics: Student versus Instructor Control over Sequencing. Unpublished Master's Thesis, University of Calgary, 1974.

Thomas, D. B. The Effectiveness of Computer-Assisted Instruction in Secondary Schools. Association for Educational Data Systems Journal, 1979, 12(3), 103-116.

Wiens, I. W. Computer-Assisted Remediation in Secondary School Mathematis. Unpublished Master's Thesis, University of Calgary, 1975.

Woetowich, H. M. A Study of Computer Assisted Instruction for Integer Arithmetic and the Effect of Some Adaptive Instructional Strategies. Unpublished Master's Thesis, University of Calgary, 1980.

# MICROCOMPUTERS IN COOPERATIVE GROUPS

Diane K. Browne and Ken Corens
1001 Highway 7
Hopkins School District 270

The microcomputer is known as an important tool in individualized instruction. The present situation, however, gives us a limited number of computers and a great number of eager students. This project proposes the use of microcomputers with groups of students using cooperative group strategies.

The initial wave of excitement that arrives with a computer often ebbs quickly when the instructor realizes the scheduling, managing, and computer operating procedures that must be conducted with the class. Grouping students on a microcomputer brings their varied listening, peer tutoring, and problem-solving abilities to the experience.

The Cooperative Learning Goal Structure assists the instructor in maintaining actively-involved students rather than spectators in the learning experience. In this way, our limited number of microcomputers can be more efficiently and effectively used to deliver the instructional program.

Participants will be presented the methodologies necessary to create cooperative lessons with microcomputer programs. Sample lessons will come from MECC or TIES Apple programs.

# MSUS/PALS: PLANNING FOR THE FUTURE

Dale K. Carrison
Dean of the Library
Mankato State University
Mankato, MN 56001

The Minnesota State University System has implemented an online bibliographic access system for its libraries, utilizing a UNIVAC 1100/80A computer. The multiple institution database, consisting of over 1.6 million bibliographic records, will be accessed by 86 terminals at ten sites when the current phase of the Project for Automated Library Systems is completed in the summer of 1981. The online catalog provides a versatility and flexibility not possible with the more static and fixed library card and COM catalogs and extends access to information resources by use of descriptive term searching and the ability to combine and limit search terms.

The last decade has brought major shifts in philosophical and operational management to most higher education libraries. The expansive environment of the past has reverted to a leveling of support and a climate of retrenchment. At the same time, pressures are exerted for new programs and increased specialization within fields of study. New teaching methodologies and emphasis on independent problem-oriented study have generated a demand for specialized information resources. Libraries must meet the continually escalating demands resulting from these factors, while still continuing to provide services to traditional academic programs.

Library development during the past decade has emphasized the creation of bibliographic data bases and products. This activity is reflected in the growing use of computerized methods of preparing traditional products, such as card catalogs, printed catalogs, and indexes. Recently, these data base products have been viewed as means for providing library users with more precise and comprehensive access to information resources. Libraries have begun to use the technology available for cataloging activities to provide information services.

Individual libraries of all sizes are recognizing that they cannot independently serve the growing needs of their many users, while realizing that the networks and utilities to which they belong can effectively meet many of these needs by exploiting their rich cooperatively built and maintained data bases. The application of computer technology to the library has made available a wide range of automated services and products, and several organizations now exist to provide automated data bases and data base products.

The concern with public access systems reflects the new philosophy that the ultimate purpose of programs for building computerized bibliographic tools is to assist the library user in locating an item or group of items that serves that user's immediate need. Traditionally, this function has been supported by print materials--the card catalog and various bibliographic publications. The first use of computers in library operations was to support the printed resources by computer production of cards and bibliographies. This development resulted in more efficient library operations, while indirectly assisting the end user in locating items in an ever-growing bibliographic universe. However, the burden of directly assisting the user with complex catalogs and bibliographies still falls largely on the information service staffs of libraries.

This background of change provided the setting for a Minnesota State University System goal of producing a plan which would maintain the excellence of System Libraries and their capacity to function in support of teaching, research, and public service. During the 1977-78 academic year, Dr. Garry Hays, Chancellor of the State University System in Minnesota, established an Advisory Committee on Libraries/Media Services with representation from all seven institutions in the System, including faculty members, Library directors, a representative of System-wide computer services, and a Dean of Graduate Studies who was designated Chairperson. The Task Force charge was to undertake a comprehensive study which would culminate in a report to the Chancellor of recommendations useful for future decision making. After many hours of general discussion, review of descriptive information on System-wide Library resources and services, and critique of draft position papers, the Advisory Committee produced, in May of 1978, a report which was, in effect, a planning document for future growth and development of information services in the State University System. The most significant parts of the report were (1) a statement of goals and objectives for State University System Libraries, and (2) fourteen recommendations relating to the future directions of Libraries in the System. These recommendations focused on continued high level of service and cost avoidance for the future, consistent with existing and projected programs and best estimates of future enrollments. Four basic recommendations, all relating to automation, formed the foundation of the report and established a basic direction which the Committee believed must be pursued if the System Libraries were to maintain a high quality of services.

In July, 1979, a decision was made jointly by State University Board personnel, the University Presidents, and the University Library Directors to proceed with a project which would focus on automation of library systems. This decision was based on the premise that a cooperative, computer-based bibliographical system would be a foundation for achievement of many of the Task Force recommendations, as well as enable each individual institution to significantly enhance its ability to meet the information needs of its students and faculty.

A conscious effort was made to involve staff from all campuses during the developmental stages of the project, which included the planning, development and implementation processes. Meetings were held with public service and technical service staff, visits were made to each library by the Project personnel, and drafts of planning documents were widely circulated. The State University System Academic and Administrative Computer Center Advisory Committees were briefed and consulted during all phases of development. The results of these activities directly affected in a very positive way the direction of Project activities. A side benefit was the level of knowledge and understanding of library automation which was achieved by participant library staff during this phase of the project; this achievement in itself was a significant Project effect.

Library automation in the Minnesota State University System has been established as an important and necessary area for development by the Chancellor and by the individual University library/media service units. Two features marked the present status of automation at the beginning of this Project in the individual University libraries: (1) the use of OCLC through MINITEX (Minnesota Inter-Library Telecommunications Exchange), and (2) varying applications of automation to purchasing and accounting processes.

OCLC operates an online computer network used by over 2,400 libraries in 50 states, Canada, and other countries. More than 3,700 remote computer terminals in the network access a very large data base (over 13.5 billion bytes of disk storage) containing information about books and other library materials. Libraries use the OCLC system to catalog books, order custom-printed catalog cards, maintain location information about library materials, and arrange for interlibrary lending of materials.

While OCLC and the purchasing and accounting systems are useful and somewhat complementary, they lack the ability to create and maintain a fully accessible online union catalog of information resources of holdings for State University System libraries, using full machine readable bibliographic records; and they do not provide an integrated system for library automation for the individual Universities or the State University System as a whole. The integrated system is needed to provide a System-wide base for automation which can complement and enhance the use of OCLC and the existing acquisitions systems, as well as provide needed services through online bibliographical access and sub-systems for acquisitions, serials control, and circulation.

The ultimate goal of the Minnesota State University System Project for Automation of Library Systems (MSUS/PALS) is automation of the existing labor intensive library operations of acquisitions, serials control, indexing access, and circulation. Automated library systems will (1) give improved accessibility to information resources for students and faculty, (2) provide management data not now accessible, (3) improve routine clerical efficiency, and (4) interface existing separate manual and automated systems at both the institutional and inter-institutional levels.

During initial development of the Project, several basic equipment alternatives, including a separate mini-system on each campus (distributive network approach), a centralized approach utilizing the existing equipment and communications network with some upgrades, and commercially available systems, were examined. The centralized approach was selected, primarily because of the high costs for equipment, facilities, and personnel in a distributive approach and the inherent consolidated database advantages of a centralized system. Commercially available systems did not meet the system requirements for online access. Since computer capacity was available on the University System academic UNIVAC 1100/80A system at Mankato, plans were developed to utilize this equipment and add mass storage and communications access as needed to accommodate the library automated systems.

A basic requirement for library automation is a computer database of current holdings which is continually updated as new materials are acquired. The construction of a System-wide

68

machine readable database, including print and audiovisual materials, is now underway at eight of the nine institutions participating in this Project (six institutions from the State University System and two other institutions which participate in the System-supported Tri-College higher education consortium; the ninth institution has no library). Each institution is responsible for development and editing of a machine readable database of all library information resources with completion of the input for basic general collections set as a goal for summer, 1980, and for the remaining special collections and formats of materials to be accomplished as soon thereafter as possible. The System-wide database of holdings will become the base for automated systems development using the sequence of (1) indexing access, (2) acquisitions, (3) serials control, and (4) circulation. The projected timeline for achievement of a totally integrated system fulfilling these four functions was originally four years--July, 1979, to June, 1983. Possible options and alternatives for Project timeline dates and activities are continually being reviewed, and the timeline updated and/or activities modified as needed.

MSUS/PALS is now in its first phase, which is development of an online catalog access system. The result of this project phase will provide an automated alternative to the card catalogs now located in the participating institutions, including both computer output microfiche and computer terminal access. In May, 1980, the access programs became operational, and tests of the access system were initiated. Test data was collected, and program modifications and revisions made as needed. The catalog access system became fully operational in its final form in August, 1980. The system was implemented for public service use on a phased basis at Mankato State, Moorhead State, and St. Cloud State Universities in the winter and spring of 1981, and will be implemented at Bemidji State, Southwest State, and Winona State Universities in the summer of 1981. During the 1980-81 year, Bemidji, Southwest, Winona, and Metro State each had one operational terminal for staff instruction and reference use.

The library system shares a UNIVAC 1100/80A computer with other State University System academic users. This host computer system includes a central processing unit, input/output unit, main storage unit and storage interface unit (total of 512K MSU and 4K SIU words) system maintenance unit, system transition unit, motor alternator, system console, FH-1782 magnetic drum, four UNISERVO 16-tape units, general communications subsystem, 0716 card reader, 0604 card punch, and 0776 printer.

The hardware configuration for the library online system includes four UNIVAC 8434 disk storage units (200,000,000 bytes per unit), two UNIVAC V77210 communications processors (asynchronous to synchronous concentrators which each support 60 asynchronous terminal ports, 4 host terminal ports, and 1 console port), and 85 ADDS R20 terminals (distributed on the basis of student population and library service access points at each institution). The library system shares ten CODEX 6010 statistical multiplexors (two for each off-site location to multiplex multiple 300 baud asynchronous lines into one 2400 baud synchronous line for terminal output and a reserve process for terminal input), and ten modems (two for each off-site location to modulate the 2400 baud synchronous line into a 9600 baud phone line for terminal output and a reverse process for demodulation for terminal input), with other academic users of the instructional computer system at Mankato State University.

The software configuration includes the UNIVAC 1100 operating system, level 36.R2B, the UNIVAC terminal concentrator (UTC) interface to the UNIVAC V77210, the UNIVAC 1100 transaction interface package (TIP), the UNIVAC 1100 data management systems (DMS) which manipulates 23 library files and related indexes, and the UNIVAC 1100 COBOL (ASCII) library system programs which include 22 batch and 12 online programs. TIP provides general transaction processing capability that allows remote terminals to initiate the immediate execution of a preregistered program and provides for specialized capabilities. This package includes TIP system program, TIP library files, CMS (communications management system) real time program which interfaces the online terminals and the system, and user transaction programs (12 library online programs in TIP).

The current online multiple institution database consists of over 1.1 million bibliographic records, of which approximately 55 percent are unique bibliographic titles. The core storage required for executing the library system is difficult to estimate because the programs are reentrant and loaded only when needed and because all programs use common bank storage areas. CMS requires 40K words, and approximately 45K to 100K words are being used at any time the system is operational. It is estimated that 6 percent of the total CPU time will be used by the library online system when it is fully operational. A statistics module records data on transactions, commands, terminal and institution usage, response time, and processing time, providing summary data online and tape output each 24 hour period.

The online catalog has five basic search strategies: author, title, combination of author and title, subject, and descriptive term. All searches are initiated by a search command consisting of either the complete word as underlined in the previous sentence or a mnemonic truncated search command consisting of the first two letters of the command word. The second part of the search command consists of the word or words which make up the search, which can either be entered as full words or entered in a truncated (shortened) form. Each of these basic searches may be limited by publication date, language, and format. Boolean connector searching, capability to save sets, and set combination by boolean logic are also available. On demand and off-line prints may be requested. The symbol # may be used to truncate words and phrases. The question mark may be used to substitute for unknown letters or words in phrases. A special module called "help" is available online to system users.

An online catalog provides a versatility and flexibility not possible with the more static and fixed card and COM catalogs. The ability to combine and limit search terms adds a dimension of retrieval power not possible in the traditional types of bibliographic access. The online catalog means a more intensive use of the library's principal resource--the collection. It also creates the possibility of extending access to the collection by means of enhanced keyword and subject indexing techniques. The online catalog presents an exciting opportunity for librarians to make a tremendous improvement in the level of catalog service available to library users.

Acceptance and orientation problems with library users when they first come in contact with the online system were anticipated. However, few of these problems actually occurred. Library users of all ages and education levels readily accepted the new "machine" which was replacing the card catalog. Most people can effectively utilize the online system after thirty minutes of instruction and use. A handbook on how to use the system is readily available. User prompts of various types are provided during the searching process. Knowledgeable staff assistance is readily available in the vicinity of all public service terminals. Several library staff have observed that patrons will, if the computer is not operational, wait for the system to become available again rather than use the card catalog. A frequent observation by students and faculty is that they are able to find information resources through the online system which they could not find in the card catalog.

A final decision on whether or not to proceed with Phase II (acquisitions) of the Project will be made after sufficient data is available to assess the results of Phase I (online catalog access) in terms of both meeting specific objectives and the impact on the capacity of the academic computer. Assuring the success of automation projects is not easy. However, carefully picking the way through the maze--assessing the library's need for automation, being a knowledgeable consumer, and adequately planning for and carrying out the system procurement and implementation process--greatly contributes to their success.

# POSITION PAPER: COMMON INFORMATION BASE FOR REGIONAL AND LOCAL DECISION MAKING

GAYDEN F. CARRUTH
DIRECTOR OF PLANNING AND RESEARCH
EDUCATIONAL COOPERATIVE SERVICE UNIT
OF THE METROPOLITAN TWIN CITIES AREA
3602 HIGHCREST ROAD N.E., SUTIE 112
MINNEAPOLIS, MINNESOTA 55418

During 1979, the Minnesota State Legislature mandated that educational cooperative service units (regional service agencies created by the Legislature in 1976) establish a common data base for local and regional decision making. The Metropolitan Educational Cooperative Service Unit, whose membership encompasses the seven county metro area, has developed a methodology and procedures for regional common data base development. This paper/presentation will detail the methodology and procedures for data base development, as well as the purposes of a common data base, data element definitions, data base description, and reporting mechanisms. The procedures being employed in the data base development process represent a unique cooperative endeavor among regional service units, regional data processing centers, and the state education agency to facilitate the availability of comparative data for local school district decision making and regional planning and research.

## Introduction

The Educational Cooperative Service Unit of the Twin Cities Metropolitan Area is one of nine regional consortia of school districts created as the result of enabling legislation (M.S. 123.58, 1976), passed by the Minnesota State Legislature in 1976. There are three primary purposes of the Metropolitan ECSU: 1) planning and research on a regional level, 2) development of cooperative programs and services, and 3) liaison with other agencies interested in education, e.g., State Legislature, Metropolitan Council, Department of Education, MECC, ESV Regions and member school districts to name a few. The Metropolitan Twin Cities encompass forty-eight school districts, each of which is required to belong to the unit for planning and research purposes.

Regional planning and research activities of the unit require the collection and analysis of school district data. Data are currently being collected in four areas: Finance, enrollment, staffing and class size. These data are analyzed and reported to school districts in the ECSU Annual Studies. As a consortium of school districts, it is the school districts themselves that approve or disapprove the collection of data for planning and research purposes. Thus we are organized to deliver our services to local districts based upon locally determined needs.

During the 1979 session, the State Legislature amended M.S. 123.58 to include the following statement in subdivision 8a:

> Insofar as possible, educational cooperative service units shall make technical assistance for long range-planning available to school districts upon request and shall establish a common data base for local and regional decision making.

The emphasis on the common data base reflects its overriding goal: Local and regional decision making.

Because data have already been collected for a number of years by ECSU and previously by ERDC, *it is the feeling of ECSU that the establishment of a common data base is part of a continuum and not a radical change in the way ECSU is conducting its data collection and analyses.* The data base that exists and is used for purposes of the Annual Studies and other ECSU research projects will be improved upon and refined to be comparable and "common" insofar as possible with data that are collected by the ESV Regions VI and VII.

The Minnesota State Legislature has mandated that the Elementary, Secondary and Vocational (ESV) Regions function as regional data processing centers with a responsibility for collecting and transmitting data requested by the Department of Education from the local districts to the Department. In order to achieve this end, a data element dictionary is being prepared by the Department. *It is the position of ECSU that the Department's Data Element Dictionary serves the needs of the Department and the State Legislature for state level policy making and that the data* *elements and corresponding definitions are not uniformly applicable to local reporting and decision-making needs. However, it is also ECSU's position that wherever possible, state level data elements and definitions should correspond to local and regional ones.* Nevertheless differences may also be necessary. Therefore, ECSU shall work with the school districts in the Seven County Metropolitan Area plus the ESV Regions VI and VII (Metro II and TIES) in the development of a common information base for local and regional decision making. The purpose of this paper is to establish a position statement on an approach to establishing such a regional common information base.

## Purpose of a Common Information Base

The purpose of the common regional data base would be threefold.

> First, it would serve as a *source of information* to be used by local school districts to make *comparisons among school districts in the region* as part of their local planning and management activities. The Annual Studies are currently being used as such a source of information.
>
> Second, it would serve as a *source of information* to be used by the ECSU to describe *regional summaries of local data* as part of the unit's regional planning and management activities. Two publications of Inquiry, scheduled for completion by spring 1981, will provide regional summaries of trends in the areas of enrollment/finance and staffing/class size.
>
> Third, it would serve as a *source of information* to be used by local school districts and the ECSU for the completion of *action research projects* recommended and conducted by the local districts of ECSU.

## Sources of Information

The data base would continue to be updated each year and would ultimately include data in the following areas:

> Enrollment
> Finance
> Staffing
> Class Size
> Facilities Utilization

In the data base would reside up to five years' data in each area. The data base could be stored on an "active" ECSU-maintained tape file. As a new year of data was added, the first year of the historical data could be dropped and stored in an "inactive" tape file maintained by ECSU

Presently, finance data are taken from the Uniform Financial Accounting and Reporting (UFARS) reports submitted to the Department of Education in August of each year. Until the UFARS data are collected in machine-readable form by the ESV regions (Metro II and TIES), ECSU will continue to request photocopies of the Annual Financial Reports from member school districts. *Once finance data were available in machine-readable form through the ESV regions, a release would be requested from each district and data would be requested from the ESV regions.*

Staffing Data is presently collected by means of a questionnaire sent to the local school districts. The questionnaire is used to collect staffing data which are comparable to salary data reported in the UFARS report. *Beginning in fiscal year 1981, facilities utilization data would be collected along with the staffing data.* Facilities data are not envisioned to be a complex set of data elements. Development would take place based on local and ESV input. Enrollment data would also be collected via the staffing questionnaire.

Class Size data are also collected by means of a questionnaire sent to the local school districts. Class size data are collected in a selected number of subject areas. Until such time that staffing, facilities, enrollment and class size data were available, data would be collected directly from the local district. *There would be an ongoing evaluation of the data base by ECSU staff in association with representatives from local districts and TIES and Metro II.* Cooperation with ESV regions to reduce duplication is discussed in a following section of this position paper.

## Data Elements

The UFARS report defines the data elements and definitions for reporting finance data to the Department of Education. The data elements would be combined by ECSU when producing local, subregional and regional summaries or when data are used in local district or ECSU related research.

Regarding staffing, enrollment, class size and facilities utilization data, ECSU would encourage local districts in association with ESV regions to establish data elements and definitions in their local data base which correspond to agreed-upon, among districts, definitions for the common regional data base. Finance and staffing data elements are identified and defined in the glossary sections of the ECSU annual finance and staffing studies. Data elements for enrollment, class size and facilities utilization are yet to be determined.

## Reporting

Data residing in the regional data base would be reported to local districts in several ways.

> First, *three descriptive reports* would be prepared in the fall of each year. The areas of reporting would be: *Finance, staffing* and *class size and school building utilization.* The purpose of these three reports would be to provide school administrators with *comparative data reported by district* for short range planning and management activities.
>
> Second, *two analytical reports* would be prepared in the fall of each year. The areas of

reporting would be: *enrollment and finance* and *class size, staffing and facilities utilization.* The purpose of these reports would be to provide board members, citizens and school district staff with *analyses of past and future regional and subregional trends* in each of the areas and also uncover relationships in these trends. This data should be useful in long range planning at the local level.

Third, *special planning and research projects* would draw upon the regional data base for the purpose of *answering questions of concern posed by local decision makers.* The results of the special projects would be reported in different ways for different audiences and would vary according to the research questions posed.

Fourth, *single district profiles* would be prepared for each of the districts in the spring of each year. Like the analytical reports they would be designed to *provide analyses of past and future regional and subregional trends in the areas of enrollment and finance* and *class size, staffing and facilities utilization.*

## Relationship with ESV Regions

In the 1980-81 ECSU Annual Plan it is stated that ECSU will:

> Coordinate the development of an accurate information base relative to education in the Metropolitan Twin Cities Area.

It also stated that in order to accomplish the goal ECSU shall:

> Cooperate with the local districts, the State Department of Education, MECC, Metro II, TIES and others in the development of a common information base in the areas of enrollment, finance, staffing, class size, and facilities utilization.

In order to accomplish this goal, *it is recommended that by fiscal year 1982, ECSU, TIES and Metro II work together on behalf of the 48 school districts in the Seven County Metropolitan area to produce the ECSU Annual Finance Study.* Specifically, it is suggested that ECSU work closely with the ESV regions in the collection of fiscal 1981 finance data to be reported in the fall of fiscal year 1982. This would eliminate the collection of finance data directly from local school districts by ECSU and should be an observable example of a cooperative working relationship. Fiscal year 1982 is the recommended year for attainment, although school districts are projected to be reporting their finance data (UFARS) to their ESV regions via the Department of Education's ESV-FIN software in fiscal year 1981.

*Regarding the remaining data elements in the areas of enrollment, staffing, class size and facilities utilization, ECSU would negotiate with the ESV regions to uniformly define the data elements residing in either existing fields or in additional fields in the local data base.* Because this would be a give-and-take process requiring ample input and cooperation from local districts and the ESV regions, it is recommended that *enrollment be completed by the summer of fiscal year 1981, finance be projected for completion by the summer of fiscal year 1982, staffing and class size be projected for completion by the summer of 1983* and *facilities utilization by the summer of 1984.* Enrollment data are presently collected by all districts and submitted to the Department of Education in the fall of each year. Data are also collected each spring by the Department of Education. Experience suggests that although enrollment is a common factor in school district funding it fluctuates by both the definition of a given enrollment category and the date the counting took place. Uniformity is not easily attained. Therefore, ECSU should work closely with the ESV regions in establishing the enrollment data elements and their corresponding definitions as established timelines for data collection. The date of implementation for this Data Base Category would be fall of fiscal year 1981 (refer to Table 1).

## Location and Maintenance of the Regional Data Base

Because the regional data base will be used primarily for planning and research purposes *the regional data base will be stored and maintained on the University of Minnesota computing system.*

The University of Minnesota at this time offers the broadest range of statistical software for ECSU use. Therefore, it is most cost effective for ECSU to maintain the University affiliation.

On those occasions where regional data may best be processed at a regional center, the selected data elements may be transferred to the region either by tape or cards. Conceived as a five year historical record of data there will eventually be two primary files in the Regional Data Base. The first (active) file representing the most current five year history. This latter (inactive) file will be created by annually transferring the oldest year of data from the active file to the inactive file at the time the most recent year is added to the active file.

This does not mean that the Regional Data Base is independent from the data bases maintained by the members of the ESV regions. On the contrary, the ideal situation is that the Regional Data Base maintained by ECSU will depend on the ESV data bases as a source of data.

Table 1
Timeline for Implementation
Region XI Data Base By
ECSU, Metro II and TIES

| Date Base Category | *Fiscal Year of Implementation | | | | |
|---|---|---|---|---|---|
| | 1980 | 1981 | 1982 | 1983 | 1984 |
| Finance | | | → | → | → |
| Enrollment | | → | → | → | → |
| Staffing | | | | → | → |
| Class Size | | | | → | → |
| Facilities | | | | | → |

* July 1 - June 30

# AN INFORMATION SYSTEM FOR MANAGING HEAD START

Linda J. Cogdill
and
Albert L. Goldberg
Wayne County Intermediate School District
33500 Van Born Road
Wayne, MI 48184

A comprehensive data management system was developed to "harness" the enormous data gathering requirements of the federally funded Head Start program in Out Wayne County, Michigan. This paper describes the conceptualization, development, and implementation of the system and outlines some of its current and future management benefits. The development of the system was difficult because of the complexity of the Head Start program requirements and the diversity of users.

Head Start was initiated in the mid 1960's as one of the panorama of "Great Society" programs. The legislation promulgating Head Start, the Economic Opportunity Act of 1964, was intended to foster the growth and development of 3 to 5 year old "low income" children and to provide a variety of parent and family support services.

Since that time, governmental data gathering for program accountability has required Head Start to develop an efficient data management system for responding to frequent local, regional, and national administrative office report requests. In the late 1970's, increased program size (1,700 families), a decline in real funding, a decrease in administrative support staff, and an increase in "paper work" resulted in the Out Wayne County Head Start program experiencing a period of data management turmoil. A computer information system was developed to organize and standardize data gathering procedures. It has relieved administrators of time-consuming and redundant accounting tasks, produced more accurate data, and provided valuable data for strategic planning.

This paper presents selected aspects of the system. It first describes the local Head Start site and operational setting and then includes a description of the data system, the rationale for the system design, and a discussion of management benefits.

## The Local Site

The Out Wayne County Head Start program began in 1964 as a small program serving 80 children. It was first administered by the Wayne County, Michigan Office of Economic Opportunity. In 1971, the Wayne County Intermediate School District[1] assumed administrative responsibility for an enlarged program serving 800 children in 8 school districts. The Intermediate School District involvement led to cooperative efforts of Title I and Head Start preschool programming and enhanced non-federal support service contributions.

With strong administrative backing, the program expanded to serve over 1,700 children in 17 school districts in 1978-79. The individual programs range in size from 25 participants served by a staff of 4 to 300 served by a staff of 25. Some school districts operate several Head Start centers; almost all provide services in more than one classroom.

The educational program designs include day care, half-day classroom participation, home visitations, and a combination of classroom participation and home visitations. In addition to the basic educational program, services to Head Start children include: health and developmental screening; health, education, mental health and social service referrals; parent involvement in program planning and administration; and assistance to families in career and life planning. The Head Start program also takes a special interest in mainstreaming the handicapped and providing a program for the bilingual/bicultural.

---

[1] The Wayne County Intermediate School District embraces the city of Detroit. Out Wayne County includes school districts exclusive of Detroit.

Funds for the administration of Head Start programs are awarded by the Federal Department of Health and Human Services, Administration for Children, Youth, and Families. The Chicago-based Region V Office of Administration for Children, Youth, and Families advises and monitors the activities of the Grantee. The Grantee, Wayne County Intermediate School District, in turn, advises and monitors the activities of the 17 Out Wayne County Head Start programs.

The Out Wayne County Head Start Program is one of the largest programs in the country. It is a comprehensive family service program, and in its diversity, complex to administer. Unusual among Head Start programs, it is administered by the public schools rather than a community service agency. While it suffers the rising costs of public school programming, it gains immeasurably from the broad range of services offered in association with public schools.

## Initial Design Considerations

The data system was developed out of the need to standardize procedures, minimize paperwork, decrease manual data-gathering activity, and yet gather sufficient information for program accounting and strategic planning.

The development of the system proved to be difficult, because a large number of users required a wide diversity of data. While the national and regional monitoring agencies required basic and periodic summary data to measure minimal performance, the local Head Start programs required detailed information on services provided to individual children. The administrative agency, Wayne County Intermediate School District, and community service agencies additionally required data which would lead to a better understanding of the program and its diverse components, indicate areas in need of bolstering or correction, and provide substantive support for future program expansion or special programming.

The goals of design were to develop a centralized data gathering system which would provide:

1. accurate and efficient retrieval of data for monitoring Head Start services,
2. statistics for reporting to the national and regional Department of Health and Human Services,
3. data for program planning with schools and community agencies, and
4. indicators of program efficacy.

Head Start programs gather data in 10 basic information categories: student information, family information, health and dental services, student evaluation, handicap services, career development planning, food program services, social services, program attendance, and parent participation. Criteria for selecting categories to be treated in a computerized data system were that the data be: objective, routinely gathered and retained, managed most easily via computer, and relatively infrequently updated.

After consideration of the 10 basic data categories using the above criteria, 5 were selected for inclusion in a computerized system: student information, family information, health information, handicap information, and student evaluation information. From the data gathered in these categories, 27 reports were designed for a network of users including monitoring agencies, private auditors, the Grantee, Head Start administrators, community agencies, and local program staff. The reports are generated at variable intervals (annually, quarterly, monthly) as the need is indicated.

## The System

Data for the computer system are gathered on 6 basic data entry forms printed on two-part carbonized paper. Data are gathered at the local program site by program staff. One copy of each form is retained at the site while the second is sent to the computer center for data entry. Data are then entered by an operator via a cathode ray tube.

The data are retrieved in two ways: data on each Head Start child are immediately accessible to the Grantee monitoring staff via the cathode ray tube; data are also available to the network of users through the 27 reports.

The system was developed as a distributive processing system to maximize growth potential. Should a distributive processing unit become available to a local school district, the local Head Start program would have the capacity to maintain files and retrieve data at the local site.

## Management Benefits

At this writing the approximate annual cost per child to maintain the computerized information system is $13.50. This figure includes computer (IBM 8100) operation, rental of cathode ray tubes, purchase of paper and forms, salaries of data entry operators, and maintenance programming costs. This cost is minimal when considering the benefits to the Out Wayne County Head Start programs.

Head Start programs are required to monitor enrollment level, eligibility based on income, and number of enrolled handicapped. These were formerly monitored by periodic spot-checking of children's files and an annual in-depth audit of selected programs by the regional office monitors. With the present computerized system there are ongoing enrollment statistics and accurate and detailed reporting for an entire Head Start population rather than merely for a sampling. In addition, enrollment and attendance history data provide information useful in recruitment and enrollment planning and program expansion.

Head Start programs are also required to monitor health services, health concerns and restrictions, and handicap services. These too were formerly monitored by spot-checking. With the computerized system, monitoring of service programs is ongoing, and reports are detailed and immediately available. As a result, programs have the ability to rapidly pinpoint children with service needs. The health and handicap services data is vital in identifying program weaknesses and is of great aid in substantiating requests for community support.

Head Start programs must respond to several lines of authority: the Regional Administration of Children, Youth, and Families, Wayne County Intermediate School District, Department of Social Services, local boards of education, and many community agencies. Prior to development of the computerized system there was no established method of gathering and providing data requested by these bodies. With the system, a vast range of information is regularly available to these interest groups, and provides a broader base for cooperative planning.

## Conclusion

The above are a few of the specific management benefits which have resulted from development of a computer information system. We strongly recommend such a system as a means of organizing and standardizing program data, relieving program administrators of time-consuming and redundant accounting tasks, and providing organizations with valuable data for strategic planning. In our case, the system has also established clear data gathering expectations of local programs and has demanded, as a function of the frequent monitoring of service to each child, that local programs provide comprehensive service programs.

The system is changing and is being refined as the network of users identify data needs for improved program management. Tasks such as improving the quality of data provided by local programs, increasing the frequency of updating, and sensitizing local programs to the value of the data are immediately important. Developing strategies for incorporating the system into existing local school district data systems, evaluating the system for new relationships among data items, computerizing data categories currently gathered manually, and forecasting possible new reporting requirements are longer term goals. Since its inception, interest in the system design and management benefits has increased among other federal programs, community agencies, and public schools. We must admit that it was not easy to develop our system or to convince all potential users of its benefits, but user response shows clearly that the product is exceeding our initial expectations.

# COMPUTER ASSISTED INSTRUCTION IN ECONOMICS: DEVELOPMENT, RESULTS AND CHALLENGES

Bruce R. Dalgaard
Center for Economic Education
University of Minnesota

M. H. Askalani
Department of Economics
Mankato State University

Paul J. Thompson
Center for Economic Education
Mankato State University

After fifteen years it is evident that C.A.I. in economics comprises a valuable mode of instruction. Acquisition of software has been a major problem as has compatability. Hardware cost problems have been removed now by the microcomputer. Efficiency dictates specialization in materials collection, cataloguing, and evaluation with subsequent sharing with others. Games and simulations seem most exportable. Mankato State has collected and converted to microcomputer many economics games and simulations. This presentment includes both a demonstration on a microcomputer and a presentation on the state of the art and effectiveness of C.A.I. in economics.

## Introduction

Computer Assisted Instruction in economics has long been of interest to the authors. Considerable advancements were made in the past twenty years to advance the use of CAI in economic education. A quick perusal of the economic education literature would seem to indicate that usage, at least as measured by published articles, peaked during the early to mid 1970s.

This presentation, which will be supplemented by a computer-based games and simulation demonstration, briefly reviews the use of CAI, summarizes one of the more extensive CAI research projects, and offers some speculative comments on the future direction for CAI in economics. The comments contained herein are meant to be part of a discussion on CAI in economics. They are not intended as a definitive statement.

## Early Historical Development of CAI[1]

Computer Assisted Instruction dates back to the late 1950s when the computer industry began using CAI programs to train their own personnel. By 1960 IBM's Coursewriter I allowed educators to do some direct curriculum programming (McLagan and Sandborgh, 1977).

Stanford University's Institute for Mathematical Studies in the Social Sciences (IMSSS) pioneered the use of CAI in an elementary school classroom in 1965 (Suppes, 1972). Over the next few years the IMSSS project made further use of the computer in classroom instruction. By the mid to late 1960s this included university-level computer-based programs.

It was also in the 1960s that the University of Illinois developed PLATO (Programmed Logic for Automated Teaching Operations).[2] In its early stages PLATO was used in library studies and nursing. CAI projects were also begun during this period at the University of Texas in chemistry, mathematics and computer programming. Florida State University utilized CAI in physics and health education courses.[3]

The use of CAI in economics began to appear in the late 1960s. One of the earliest reported experiments was by Moncreiff at Teachers College of Columbia University (Moncreiff, 1965). In a review of CAI research in economic instruction contained in K. Lumsden's 1967 publication, New Developments in the Teaching of Economics, T. Rodgers reports on four such studies. Most were computerized versions of programmed learning materials.

Since the late 1960s hundreds of CAI experiments in economic education have been undertaken. As Booms and Kaltreider pointed out, it was usually "dissatisfaction with typical large class instruction...(which prompted attempts) to find a system for teaching large classes...." (Booms and Kaltreider, 1974, p. 408). CAI was often seen as a method to increase learning and improve attitudes. In a 1974 review of CAI in economics, Soper concluded that "it would seem fair to say that the discipline has made reasonable progress in the development of materials for computer-assisted instruction" (Soper, 1974, p. 23). Indeed it did, at least into the mid 1970s. The progress achieved from the 1960s to the mid 1970s did not seem sufficient to generate increasing use of CAI in economics, at least not at the college level.

One additional conclusion from Soper's review article may help explain this assertion. Soper noted that "the quantitative evidence bearing on the effectiveness of CAI is presently not compelling...." (Soper, 1974, p. 24). Subsequent CAI research seems to make this conclusion obsolete. One such research study is reported below.

## Computer Assisted Instruction - Study Management System[4]

During the past decade the computer has been employed in the teaching of the principles of economics course using PLATO - the University of Illinois' version of CAI. Six instructional sequences and an elaborate testing and record-keeping system have been developed. During one four-semester period, a student "study management system" was also used in conjunction with the foregoing instructional sequence and the testing routines. The results of this varied program are reported herein.

### Part I. A study management system.

With the cooperation of the Department of Educational Psychology, a so-called "study management system" was developed for the first course in economics (Anderson, T. et. al., 1974). During the first year of the project 228 students were assigned randomly between PLATO classes and non-PLATO classes (two each). Both groups used the same text, both studied with the same instructors (each instructor had a PLATO and a non-PLATO class), and all students took the same examinations. Students in the two CAI classes were required to go to terminals in the library where, upon signing in, they were given a short reading assignment from the text and were then automatically signed off the system. Upon completing the reading assignment they returned to the terminal and took a short quiz over the materials studied.[5] If they achieved a specific level of proficiency, they were given another assignment and tested, and the process was repeated. If unable to demonstrate proficiency, students were instructed to restudy the assigned materials prior to retaking the quiz.

Inasmuch as students could not take graded hour examinations without completing the PLATO quizzes, the process insured that students attended to their "homework" with reasonable faithfulness. "Cramming" was discouraged by limiting access to the terminals. Expectations of positive results from this process were enhanced by some 70 years of research indicating that activities which increase meaningful cognitive processing on the part of students significantly improve learning. The technique of asking students questions (as with the PLATO quizzes) was regarded as one such activity (Anderson, R., et. al., 1975).

The statistical design of the study and the conduct of the course have been reported upon at length elsewhere (Anderson, T., et. al., 1975). The outcomes in terms of achievement, however, are recapitulated briefly in Table 1.

Table 1

| Examinations | Mean Scores<br>CAI<br>(n=79) | <br>Non-CAI<br>(n=99) | Standard Deviations<br>CAI | <br>Non-CAI |
|---|---|---|---|---|
| First hour | 23.7 | 22.5 | 3.34 | 3.30 |
| Second hour | 32.0 | 30.2 | 4.91 | 5.89 |
| Final | 23.0 | 21.7 | 3.72 | 4.54 |
| Standardized | 17.1 | 16.6 | 3.25 | 3.48 |

On the first and second hour examinations and on the final examination, students using CAI did significantly better than the control group.

Course attitudes for students using CAI instruction (as measured by a standardized Course Evaluation Questionnaire) were significantly better than for the control group. Indicative of their opinions, 90 percent of the CAI students reported that the study management system "helped them learn more." In the control group 23 percent who enrolled (or attended class and could have enrolled) did not complete the course. Possibly some of the students assigned to the CAI sections - many of whom were freshmen - may have been frightened initially by the prospect of using the computer. A multivariate analysis of variance involving sex of student, high school rank, cumulative college hours, college grade point average, and year in school did not show a significant difference between students who dropped the course and those who did not.

In summary, the study management system used in this experiment resulted in students, overall, learning significantly more (as measured by the testing instruments used) and in improved attitudes toward the course. As judged by questionnaire answers, students perceived they learned more using PLATO because they studied with greater regularity, studied "harder," and because they were required to demonstrate understanding of important concepts.

These findings agree with another study involving Computer Assisted Instruction called TIPS (Teaching Information Processing System) (Kelley, 1968), conducted by Allen C. Kelley. Both the study management system described here and TIPS attempt to "direct" student activity. Both systems use "surveys" or "quizzes" to focus student attention on important concepts. Both systems provide students with prompt feedback concerning their progress in the course and the possibility of special attention depending upon their performance. Both systems are reasonably economical of computer time. And both systems report statistically significant improvement in student performance and attitudes.

### Part II. CAI testing, seminars, and student tutors.

The second year of the project was devoted to building upon the favorable outcomes of the study management system. Except for a relatively small lecture-discussion class used as an informal control group, lectures in the course were discontinued in favor of an initial three-week orientation period (using quiz sessions), followed by a number of three-week seminars on a variety of topics of current interest - one of which students were required to participate in during the course of the semester. Three one-hour examinations, which students had the option of retaking in other forms, were administered by PLATO on a flexible schedule. Students were also encouraged to elect an early pencil-and-paper final examination, thereby optimally finishing the course before the end of the semester. Throughout the semester, of course, students read the text and took quizzes administered via PLATO on the assigned reading. Except for the necessity of finishing given blocks of material before taking (and possibly retaking) the one-hour examinations, students could proceed at their own pace.

Although statistics on performance for the large CAI group and the students in the small, informal control group were incidental to the experiment, they are reproduced below for purposes of reassurance (Dalgaard, et. al., 1975). Only the results of the final examination are available for comparison (see Table 2).

Table 2

| Final Examination | Mean Scores CAI (n=318) | Mean Scores Non-CAI (n=64) | Standard Deviations CAI | Standard Deviations Non-CAI |
|---|---|---|---|---|
| Text-related questions | 20.45 | 18.86 | 3.55 | 4.15 |
| Standardized questions (JCEE) | 20.52 | 20.61 | 4.11 | 4.98 |
| Graphic models | 6.19 | 6.97 | 1.83 | 2.25 |
| Total Test | 46.52 | 45.43 | 7.87 | 9.79 |

As perhaps might have been expected, the CAI students did significantly better on text-related items; there was no significant difference on questions from a standardized test of the Joint Council on Economic Education; the non-CAI group scored significantly higher on questions related to graphic models (over which they had more drill in class than the CAI students). Overall there was no significant difference between the two groups - a finding confirmed by regression analysis. What is important in this "non-experiment," however, is that the study management system apparently served as an effective substitute for lectures and for typical quiz sessions. The computer seemed to be an effective substitute for the instructor of the control group.

Several related and important aspects of the experiment are worth noting (Anderson, T., et. al., 1975 - II). First, the administration of the examinations and the related record-keeping chores performed by PLATO enormously simplified teaching for the instructors. Moreover, of students expressing a preference, 70 percent thought the on-line tests were superior to pencil and paper examinations. The opportunity to retake examinations was thought to be helpful or very important by 78 percent of the students. Instant grading and the ability to go over wrong items at the students' leisure were clearly desirable characteristics of the system.

Second, for the instructor, the availability of tests comprised of items of proven validity and the automatic keeping of records took a great deal of the drudgery out of teaching a large section of the elementary course. During this part of the experiment the examinations were proctored, although they have also been used in an unproctored setting with reduced weights.

Third, the seminars were a successful innovation, making it possible for students to study in detail one or more areas of applied economics. Meeting in relatively small groups (15), students had an opportunity to relate to instructors on a face-to-face basis. Fifty-six percent of the students thought the seminars either very important or helpful.

Finally, the use of undergraduate student tutors was highly successful, both from the standpoint of the students being tutored and the tutors themselves. In terms of rapport with students, interest in the course, familiarity with PLATO, and faithfulness in keeping office hours, the undergraduate tutors did at least as well as the teaching assistants. More than 50 percent of the students in the course commented favorably on the tutors - a surprisingly high percentage since many students had no occasion to contact them at all.

### Summary of Caisms

Here is an instance of a carefully conducted series of studies reporting on the role of the computer in the beginning course in economics. The first year of the experiment (Part I) demonstrated that for randomly assigned groups of students a study management system facilitated learning. The rationale was that such a system would require students to attend to homework on a regular basis and that computer-administered quizzes would increase the probability of students engaging in meaningful cognitive processing while doing so.

The second year of the project (Part II) demonstrated that a study management system, when coupled with examinations administered by the computer, seminars, and undergraduate tutors, was an effective substitute for formal lectures and most discussion sessions. Under such a regime students did as well as students taught by more conventional methods.

### Implications and Conclusions for College-level CAI

If the above study can be used as an indicator of the effectiveness of CAI in large-scale, college economics instruction, then one might reasonably conclude that CAI is in widespread use in post-secondary economics instruction. Actually, if research and evaluation reported in the Journal of Economic Education is a measure of activity, the quantity of college CAI economic education has dropped. There have been virtually no research studies reported since the 1977 article summarized above.[6]

Perhaps Soper's other conclusion, that CAI "is expensive and will probably remain so," explains why the reported use of CAI in college-level economics has not kept pace with expected levels, given the expansion of the late 1960s and early 1970s. With the current technologies reducing the cost of acquisition and implementation, this barrier should be eliminated.

Nevertheless, does the relative absence of reported computer-based research in economic education mean that CAI is not being used in economics? Not quite.

The recent development of an economics-based videodisc-computer project by the Minnesota Educational Computer Consortium (MECC)[7] demonstrates continued interest in the use of the computer in economic education. And the future looks bright. The orientation may well be toward the elementary and secondary school classroom rather than the college level.

## CAI in Secondary/Elementary Schools

The use of CAI at the secondary and elementary school levels is expanding rapidly. A recent survey assessed the current and projected use of the computer in U.S. public secondary and elementary schools, with special emphasis on the use of CAI (Chambers and Bork, 1980).

The authors reported that 74% of all districts responding to their survey use the computer in support of the instructional process and 54% utilize computer assisted instruction. Furthermore three-fourths of the districts responding to this national survey indicated that their faculties are interested in developing new CAI programs.

While the survey does not report CAI usage by course taught, it does report that CAI is used most extensively in mathematics (by 54% of the districts) with business and social sciences combined second (39%). In projections for the 1980-85 period, the study indicates that nearly 55% of business/social science departments plan to utilize CAI.

The study concludes that in the next decade "cost reductions due to mass production and consumption for home entertainment and learning will permit cost/effective uses of computer assisted learning in both the traditional classroom and in other settings" (p. 5). The mini and micro computers seem to be the vehicles for even heavier CAI in the secondary and elementary schools.

## The Remaining Challenge

Now that the equipment affordability question seems to be on the verge of solution, attention needs to be focused on two other issues, teacher acceptance of the computer and availability and transportability of programs. The survey cited above indicates that elementary and secondary level teacher acceptance may no longer be a barrier - "overall, faculty and students are positive in their views of computer assisted learning as an instructional tool, and over three-fourths of the districts indicated faculty are interested in developing new computer assisted learning programs." (Chambers and Bork, 1980, p. 15).

Probably the most difficult problem still facing economic educators is the availability of good CAI programs in economics. Until recently commercially developed CAI materials in economics have been very limited.[8] While it is evident that many economics teachers have developed and used computer programs for teaching economics, awareness of what exists has been limited. To further complicate matters, existing CAI materials have been programmed in a variety of different languages for use on a variety of types of equipment. This makes it virtually impossible for a non-specialist to acquire materials and translate them into a format which is usable on the particular equipment available in a given school. Even the specialist must undergo substantial costs in time and resources to make the necessary conversions. Only the most dedicated will endure.

Again, technology may be paving our way to future success with computer assisted instruction. Given the affordable prices of micro-computers, thousands of different schools can be expected to own a given brand of computer. Schools which use the same type of quipment will then be able to duplicate and exchange CAI programs with one another at a very nominal cost in terms of both time and financial resources. One more example from Minnesota will illustrate this point.

The Minnesota Educational Computer Consortium has made the decision to provide MECC members with software for one brand of micro-computer. Programs already available on MECC's big, time-share computer are being translated into a format that is usable on this micro-computer. Schools that are affiliated with MECC can load these programs onto their micro-computers either by means of telephone connection to the central time-share computer or by ordering pre-programmed floppy disks which are mailed to the schools.

The future looks bright for computer assisted instruction in economics. The authors' optimism is based on the expectation that a proliferation of computers will bring a proliferation of economics program writers. Until that happens, it is essential that we gain a collection of those materials which have already been developed. Furthermore, these materials need to be catalogued, evaluated, and publicized so that prospective users might be made aware of what is available.

Through the efforts of the Minnesota State Council on Economic Education and the Mankato State University Center for Economic Education a substantial amount of energy has been devoted to this task. A variety of economic education computer programs have been collected, many have already been evaluated, and the array of programs is being publicized throughout the economic education network.

A selection of these economic education programs, especially those suited for the micro-computer, will be presented during this session.

## Conclusion

The use of computer games and simulations and computer assisted instruction in economics has demonstrated the viability of these instructional tools. Research results support the contention that CAI can be successfully used to complement, and that a CAI-SMS system can be used as a supplement for traditional college-level economic instruction. Technological breakthroughs are making interactive computers an affordable part of the instructional facilities at most colleges and many elementary and secondary schools. With student and teacher attitudes being positive, the opportunities for the use of computer-based instruction, especially at the pre-college level, are tremendous. Time, effort, and resources need to be devoted to making existing and newly developed programs adaptable and transportable.

## FOOTNOTES

[1] See Suppes and Macken, 1978 for a more detailed discussion of this topic.

[2] A more extensive discussion of one application of PLATO is presented later in this paper.

[3] A brief discussion of major CMI and CAI systems can be found in Educational Technology, January 1979.

[4] For a more detailed version of this discussion see Paden, Dalgaard and Barr, 1977.

[5] There were approximately 300 questions on the PLATO quizzes which were answered by the experimental group of students. These covered 25 chapters of the Samuelson text.

[6] One exception is a follow-up study related to CAISMS by Paden and Barr (1980).

[7] The Minnesota Educational Computer Consortium (MECC) is an example of such a time-sharing system. Minnesota schools have teletype access to the MECC computer and its library of programs. However, most economics teachers seem to have been reluctant to use the MECC system, probably because these teachers felt uncomfortable with the equipment, but also because the teletype terminals were nearly always under the physical control of math and science teachers who were reluctant to share their monopoly control.

[8] Copyright problems may hamper the development of commercial programs. It is extremely easy to duplicate programs that are available in floppy disk form. This may reduce the potential profits from the sale of such programs as teachers will be tempted to give away programs to their friends.

## REFERENCES

Anderson, Richard C., and W. Barry Biddle, "On Asking People Questions About What They Are Reading," in Psychology of Learning and Motivation, Gordon H. Bower (Ed.), Vol. 9. New York: Academic Press, 1975.

Anderson, Thomas H., Richard C. Anderson, Bruce R. Dalgaard, Edward J. Wietecha, W. Barry Biddle, Donald W. Paden, H. Richard Smock, Stephen M. Alessi, John R. Surber and Laura L. Klemt, "A Computer Based Study Management System," Educational Psychologist, 11 (1974), 36-45.

Anderson, Thomas H., Richard C. Anderson, Bruce R. Dalgaard, Donald W. Paden, W. Barry Biddle, John R. Surber and Stephen Alessi, "An Experimental Evaluation of a Computer Based Study Management System," Educational Psychologist, 11 (1975), 184-190.

Anderson, Thomas H., Richard C. Anderson, Stephen M. Alessi, Bruce R. Dalgaard, Donald W. Paden, W. Barry Biddle, John R. Surber, and H. Richard Smock, "A Multi-faceted Computer Based Course Management System," in Proceedings of the Second World Conference on Computers in Education. Amsterdam: North Holland Publishing Company, 1975.

Askalani, Mohamed and Paul Thompson, "Computer Assisted Instruction in Economics," unpublished manuscript, October 1980.

Booms, Bernard H. and Kaltreider, D. Lynne, "Computer-Aided Instruction for Large Elementary Courses," American Economic Review, May 1974, 408-413.

Chambers, Jack A. and Alfred Bork, "Computer Assisted Learning in U.S. Secondary/Elementary Schools," Center for Information Processing, California State University Fresno, Report #80-03, July 1980.

Dalgaard, Bruce R., Donald W. Paden and Thomas H. Anderson, "Computer Based Course Management in the First Course in Economics," Proceedings of the Fifth Annual Meeting of the Illinois Economic Association, Chicago, 1975.

Kelley, Allen C., "An Experiment with TIPS: A Computer Aided Instructional System for Undergraduate Education," The American Economic Review, 58 (1968), 446-457.

Lumsden, Keith G., ed., New Developments in the Teaching of Economics, Englewood Cliffs, N.J.: Prentice-Hall, 1967.

McLagen, P. A. and R. E. Sandborgh, "CAI's Past: We've Come a Long Way," Training, September 1977, p. 52.

Moncreiff, B., "The Sumerian Game: Teaching Economics With a Computerized P.I.," Programmed Instruction, 4 (1965), 10-11.

Paden, Donald W. and Michael D. Barr, "Computer-Assisted Instruction in an Elementary College Economics Course," Computers & Education, Vol. 4, #4 (1980), 259-267.

Soper, John C., "Computer Assisted Instruction in Economics: A Survey," The Journal of Economic Education, Vol. 6, #1, Fall 1974, 5-28.

Suppes, Patrick and Elizabeth Macken, "The Historical Path From Research and Development to Operational Use of CAI," Educational Technology, April 1978, 9-12.

Suppes, P., "Computer Assisted Instruction at Stanford," Man and Computer, Basel: Karger, 1972.

# TI LOGO: A TURTLE OF A DIFFERENT COLOR

John D'Angelo
Corporate Engineering Center
Texas Instruments Incorporated
P. O. Box 225474, MS 370
Dallas, Texas 75230

Children are growing up in a time where the computer is common-place. LOGO, a language designed for children, is an appropriate vehicle for giving children the opportunity to explore the world of the computer. Although the history of LOGO goes back more than twelve years, it has not been until the recent implementation of the micro-computer that it has had such wide exposure. The basic premise in the use of LOGO, is that the child programs the computer, rather than the reverse. The computer becomes readily available to the child, just as his pencil is always readily available. The child uses it throughout the curriculum, applying it as appropriate. This paper will also describe LOGO in use in the Lamplighter School in Dallas and its use in the New York City Schools.

## I. Historical Background

LOGO is one the many Actor Languages which have appeared on the computer scene over the past ten to fifteen years. Actor Languages can be summed up as follows:

> "Actor languages are customizable and extensible, with particular power for graphics. Changes and additions in command structure can be made easily and rapidly."[1]

Although the name Wallace Feurzieg has been linked with the origination of LOGO, it was Seymour Papert of the MIT Artificial Intelligence Lab, who developed it fully over the past twelve years. He, and his colleagues at MIT have been guided by the principle; "No threshold; no ceiling". Originally implemented on large mainframe computers, it was considered a major break-through when it was implemented on a smaller, less costly computer, the DEC LSI11. At that point, LOGO could be manufactured and sold at a price of $15,000 for one system. This did not encourage many schools, the prime area of interest, to beat down the doors of MIT to call LOGO their own. This is not to say that LOGO was not used in schools, for it was throughly tested in schools over the last twelve years. However, at that price, schools could not easily afford LOGO. In 1978, an agreement was reached between MIT, Texas Instruments' Corporate Engineering Center, and a private school in North Dallas to develop a version of LOGO to run on the, then new, TI microcomputer, the 99/4. The first of this implementation was introduced in the Lamplighter School in November of 1979. This consisted of a standard TI 99/4 and a "black box" memory extension. The success and positive response was immediate.

## II. Early Usage

An early LOGO use was in Brookline, Ma. This was an 18-month study of an elementary school computer laboratory. The study, funded by the National Science Foundation , and carried out by the LOGO Group at MIT in collaboration with the Public Schools of Brookline, provided fifty students in the sixth grade with between 24 and 36 hours of computer time and instruction in LOGO. From the total group, a portion of these 16, were selected to represent a wide range of abilities, interests, and learning styles. These sixteen were monitored thoroughly, analyzing them in the areas of computer programming, mathematics, and problem solving.[2,3] These studies are still obtainable from the MIT LOGO Group.

One of the more significant uses of LOGO prior to the Texas Instruments' 99/4 implementation, was its use at the Cotting School, a private high school for cerebral palsey children in Boston. The successes and triumphs of these children have been documented in variouis MIT publications and reports, therefore, I will not discuss it here. It is interesting to note that of the five seniors who used LOGO and have graduated, all have chosen computing as careers.

## III. The LOGO Environment

One is first tempted to describe LOGO as a computer language, but this falls very short of an accurate description. Rather than a language, LOGO is a learning environment. It is an entire system for learning. It is an instructional delivery system. As I have pointed out in an earlier article[4], the present delivery system has reached maturity. With LOGO we have an opportunity to introduce a new tool, a new vehicle, a new process for delivering the products of education.

Dr. Seymour Papert contrasts the previous delivery system's attempt to use the new computer technology in this way.

> "Much of the work done in the whole area of computers and education-e.g.: CAI-has promoted a style of learning that gives the impression of a child being "programmed" by the computer. Our approach has been diametrically opposed to that. By striving to make the computer's processes as transparent as possible and creating activities in which children "teach" (i.e.: program) computers in a well-structured, procedural language like LOGO, we have aimed toward putting children in control of their own learning.[5]

A critical issue facing us today is: who will control the computer? Children feel very comfortable with computers for they have grown up with them. Using microprocessor-based toys or games and video games, the child can hardly miss using a computer almost daily. Dr. Dan Watt, Director of the Computer Resource Center at Technical Education Research Center, Cambridge, points out that: "Who is in control of the computer may be one of the critical issues of the next few years. I believe that citizens of the 1980's and 1990's must understand the ways they can control computers and the ways in which computer systems and programs can be and are being used to control and manipulate them."[6] The LOGO environment gives the child the opportunity to control the computer. This environmnent needs to be a free, open one, with ready access to "his" computer. We must think of the computer as we think of the pencil, always available, to doodle with, write poetry with, or with which to solve partial differential equations. This, the LOGO environment permits.

## IV. Using LOGO

LOGO, as a programming language, has been designed as an easy-to-use, and to learn, interactive language. While especially designed for children, it has been used at all levels. When children first sit at the computers, with very little instruction, they can immediately interact with their computer and cause it to do something they want it to do. For example, if one were to type in FORWARD 50, a line of 50 units would be drawn on the screen. By telling it to RIGHT 90, gives a ninty degree turn. Repeating that four times gives a box. All this is achieved with a minimum of instruction. They can just as easily draw triangles. However, a FORWARD 50, RIGHT 60, as we usually think of the angles of a triangle, creates a hexagon, much to our surprise! Now comes the first opportunity for new learning. Why does this happen? We then put the learner in the place of the object drawing the line, called the turtle for traditional reasons. If the child were to be the turtle, how far must one turn to walk the path of the triangle? One soon discovers that it is necessary to turn 120 degrees and not 60, for one is walking an outside angle. A new way of looking at geometry emerges.

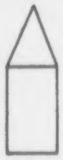
One then can "teach" the computer a new word. It knows how TO FORWARD and TO RIGHT, but now we can teach it TO BOX as follows.

```
TO BOX
REPEAT 4 [FORWARD 50 RIGHT 90]
END
```

That simple procedure will draw a box each time BOX is typed on the computer. To draw a triangle we teach it TO TRIANGLE.

```
TO TRIANGLE
REPEAT 3 [FORWARD 50 RIGHT 120]
END
```

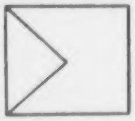
The LOGO learner now has written two procedures. (In LOGO, we call them 'procedures', rather than programs.) We can now combine these into a new procedure which would draw a house.



If we write the procedure

```
TO HOUSE
BOX
TRIANGLE
END
```

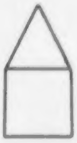
we end up with a house which looks like this:



It obviously has a "bug" in the procedure. This is a very powerful concept for the learner, for we do not say that the procedure is wrong, but only that it has a bug in it. Most children are not accustommed to this. They are usually told that they are wrong and that they are to do it right. But in this case, they are asked to "fix the bug". The TRIANGLE procedure needs to be shifted upward and turned. One way they might do this, and there are other ways, is:

```
TO HOUSE
BOX
FORWARD 50
RIGHT 30
TRIANGLE
END
```

and it looks like-



Here we see a debugging process and the use of sub-procedures, all within the first brief introduction to LOGO. The commands are simple and straight-forward. The programming is natural and almost unnoticeable. They are in command of the computer.

To further illustrate the power of LOGO as a process, think of the circle. Using the Turtle, how would one draw a circle. We ask the child to BE a Turtle and walk in a circle. How does one do that? Walking, one goes forward a very little, then turns a very little; repeating that until you have walked in a circle. We find that in LOGO it looks like this.

```
REPEAT 360 [FORWARD 1 RIGHT 1]
```

That shows, with very little doubt, the circle as a process.

Another whole realm of LOGO is the use of "sprites", mechanisms (or actors) within TI LOGO to give the user access to shapes, motion, and the capability to create dynamic pictures; i.e., movies. By telling one of 26 sprites about itself, it is, for the child, a fantastic creation limited only by imagination. One can tell the sprite its color, speed, heading, and what shape its carrying. The child can use one of the six shapes built into LOGO, or can create a new shape of virtually any design. Unfortunately, since I cannot show the colorful, dynamic quality of this aspect on the printed page, it will have to be left to your imagination or you must wait until the next available demonstration.

## V. Microcomputer Usage

As a result of the collaborative efforts between MIT and Texas Instruments, LOGO has been implemented on the TI 99/4 microcomputer. The project uses unmodified, off-the-shelf microcomputers with a memory extension. Presently 50 computers are available to 300-400 children. The Lamplighter School has pre-schoolers (3-5 year olds) through 4th graders. Pre-school through 1st grade use LOGO, as an instructional aid, rather than for programming. In this instance, theachers have developed a series of programs which provides color recognition, block building, shape comparisons, and motion coordination. Even these children use LOGO on a daily basis.

Children have virtually free access to the computers, since they are located in the classrooms and in the adjoining shared spaces. Children use any available time to use the computers including coming early to school and staying quite late.

LOGO is also presently used in the New York City Public Schools through a project sponsored by the New York Academy of Science. One or two LOGO configured computers are located in several schools in Manhatten and the Bronx. We have noticed the same level of enthusiasm at both the Lamplighter School and the New York Schools.

## VI. Applications to Learning

Relative to the teaching and learning of mathematics, we find ourselves in a difficult position, Edsger Dijkstra points out that "Since the turn of the century the American culture has gradually turned itself away from mathematics (see Lynn A. Steen, "Mathematics Today", or Morris Kline, "Mathematics in Western Culture") to the extent that many of its otherwise well-educated citizens have no longer a balanced view of what mathematics is all about. The culture has become amathematical, sometimes even antimathematical."[7] LOGO has the potential to greatly modify this attitude.

There is a school of thought which contends that general methods for solving problems should be taught. Seymour Papert, in his book, Mindstorms, mentions that in order to solve a problem, we need to subdivide it into simpler parts.[8] This was true in the discussion earlier when we described the process one needed to go through in order to draw a house. Turtle geometry allows us the opportunity to do this with little difficulty. While we are talking at the abstract level when we say that to solve a problem we compare it to that which we already know, in Turtle geometry terms we -Play Turtle. That is to say, we look to our own experiences, our own behavior and replicate our body motions. We apply those same processes. With repeated success using Turtle geometry, the process becomes part of the child. The school mathematics curriculum as it is presently constructed, does not allow children to draw upon their own experiences, and is therefore, a poor model to use for learning problem solving.

Let us recall the process used earlier to draw a circle with Turtle geometry. This is very unlike what we learned as a description of a circle using Euclidean or Cartesian geometry. In these cases, we are speaking of a circle as the loci of points equidistant from a point or as an equation $A^2 + B^2 = R^2$, either of which make little sense to the learner who is encountering a circle for the first time. Whereas, if we describe the circle as FORWARD 1, RIGHT 1, the child gets an immediate "feel" for the circle. A whole new concept for learning geometry is introduced.

While it is true that the use of LOGO assists with learning problem solving, enhances logical thinking, and makes geometry more relevent to the child, many educators are more comfortable if they see where it relates to the present curriculum in terms of a specific subject matter. If I am forced to link it to a particular subject then, I would have to see LOGO as an excellent vehicle for that which we have come to describe as computer literacy. Dan Watt describes computer literacy "as that collection of skills, knowledge, understandings, values, and relationships that allows a person to function comfortably as a productive citizen of a computer-oriented society."[9] Using a child-appropriate computer language such as LOGO, the learner is put in control of the computer immediately and is taught to expect to be in charge. Later in life, such a person will object strenuously if put into a situation where the computer attempts to run some aspect of that person's life. Currently many individuals who were brought up without any interaction with the computer are victims of computer control. They have no experience base on which to draw which reminds them that they could be in control. Todays children do have that opportunity. Just as literacy in a language means communicating by using the ability to read and write, and literacy in mathematics means the ability to solve problems, not simply add and subtract numbers, the ability to control and, even in some simple way, to program a computer is a key element in any computer literacy program implemented by a school system, and is a required skill for a person to be considered "computer literate".

All of the above aside, I consider the place of LOGO in the curriculum at about the same place as one would put the Pencil!

---

[1] Nelson, T., "Symposium on Actor Languages", On Computing, October, 1979.

[2] "An Assessment and Documentation of a Children's Computer Laboratory", LOGO Memo 48, 1977.

[3] "An Interim Report of the Brookline LOGO Project", LOGO Memo 49, 1978.

[4] D'Angelo, J., "The Microprocessor as Pencil", AEDS Monitor, Oct/Nov/Dec, 1979.

[5] Papert, S., "New Cultures from Old Technologies", Byte Publications, Inc., Sept. 1980.

[6] Watt, D., "Computer Literacy: What Should Schools Be Doing About It?", Classroom Computer News, Nov/Dec 1980.

[7] Dijkstra, E.W., "An American Programming Plight", ACM Sigsoft, Software Engineering Notes, Jan. 1981.

[8] Papert, S. "Mindstorms, Basic Books, 1980.

[9] Watt, D., "Computer Literacy: What Should Schools Be Doing About It?", Classroom Computer News, Nov/Dec, 1980.

# THE DEVELOPMENT AND OPERATION OF A PROGRAMMING COOPERATIVE

Paper Prepared for Presentation at AEDS-81

By: Arthur W. Daniels, Jr.
Executive Director
RAMS Cooperative
31201 Dorchester
Madison Heights, MI 48071
(313) 585-7530

## ABSTRACT

The RAMS Cooperative was formed in 1974 to serve the advanced application development needs of the Oakland and Wayne County Intermediate School Districts in the State of Michigan. The RAMS Cooperative was expanded in 1978 to include the Ingham, Saginaw, and Washtenaw Intermediate School Districts as well as Northern Michigan University (located in Marquette, Michigan, and serving the local and intermediate school districts of Michigan's Upper Pennisula). The Muskegon Area and Calhoun Intermediate School Districts joined the RAMS Cooperative in 1980. The participation in the RAMS Cooperative is voluntary, not mandated.

The history, development, and operations of the RAMS Cooperative will be discussed.

The RAMS Cooperative was officially formed on February 1, 1974. This official action took place at a meeting held in Ann Arbor, Michigan, that comprised the data processing staff's of Oakland Schools and Wayne County Intermediate School District, and additionally those staff members of Wayne and Oakland that were to become staff members of the RAMS Cooperative. The official action was the formal signing of the RAMS Cooperative Agreement by Dr. William Emerson, Superintendent of Oakland Schools, and Dr. William Simmons, Superintendent of Wayne County ISD. The signing had all of the pomp and cermony of a presidential approval. And, I am happy to say, as of February 1, 1974, everything was on schedule.

In actuality the effective date of the formation of the RAMS Cooperative was January 1, 1974, and some members of the RAMS Cooperative staff were already in place, namely myself and a secretary. The other members of the RAMS Cooperative, as I indicated, came from the data processing staffs of Oakland and Wayne, and they retained their employment status with their original employer. Needless to say, this caused me considerable headache for quite some time, trying to administer a then 16 person staff operating under 3 sets of personnel policies.

I submit to you, however, that the actual formation date of the RAMS Cooperative was long before 1974, and I would further submit that I can only be credited or blamed for the progress to date since 1974. I believe the RAMS Cooperative was really formed in 1965, and although it didn't have the name RAMS then, I believe the wheels were set in motion then.

Dr. William Emerson has always received a great deal of credit for the excellence of Oakland's data processing installation, and for his leadership role within the County of Oakland, State of Michigan, and the United States of America. Dr. Emerson had a unique talent of encouraging people to strive for excellence and to try a little harder, and because of the admiration and respect his people had for him, many otherwise average people accomplished great things. I submit, however, that great people can only achieve their great success through the efforts or others, and Dr. Emerson certainly surrounded himself with talented individuals.

Yes, Dr. Emerson has always been and will always be one of my idols, and I was greatly saddened by his death in February 1980. However, although Dr. Emerson has received most of the credit for the formation of the RAMS Cooperative, it was two other individuals who got their heads together in 1965 or 1966, and probably at a MAEDS meeting, to lay out a programming cooperative venture to meet the then data processing needs of metropolitan Detroit area school districts, namely Wayne, Oakland, and Macomb County ISD's and the City of Detroit Schools. By this time Dr. Rex Wood's doctoral thesis entitled "A Telecommunications Network for Data Processing in Schools: A Mathematical Model," was already published, and it served as a basis for the discussions among Mr. Ebersole, who was then Director of Data Processing for Detroit Public Schools, on loan from Wayne County ISD and who served MAEDS as President in 1965 and Convention Coordinator for the 1967 AEDS Convention, and Rex Wood, who was then Director of Administrative Services for Oakland Schools and had served MAEDS as President in 1964. The result of their discussions was the submission of a funding request to the Federal Government for the formation of a Programming Cooperative, and after a couple of revisions funding in the amount of $650,000.00 was realized for a two year period of 1967 through 1969. This Federally funded project was referred to as Integrated Educational Information System (IEIS). The project was administered by Macomb ISD as fiscal agent, and the staff was located in a vacant store in Warren.

One may reflect that those were the days (1967) of considerable concern over the possibility of forced busing, and understanding the high tolerance level of the residents of Warren, Michigan, for innovational educational improvements such as forced busing, it necessitated the sign on the door be changed from Integrated Educational Information Systems to simply IEIS.

Committees of local district users, data processing staff personnel from Macomb ISD, Oakland Schools, Wayne County ISD, and Detroit Public Schools, along with staff members of IEIS, met regularly to develop detailed system and program specifications. The functional specifications had already been defined by Dr. Wood's thesis.

Due to an outstanding marketing effort and the confidence that the future could only get better, IEIS selected a dynamic, new programming language that would allow for rapid programming, easy testing and debugging, and a high degree of maintainability. The programming language selected was PL/1. It did not matter that PL/1 might perform just bit slowly on hardware, because time was short and there was a great programming effort to be accomplished.

Generally, I believe all will agree that the programming objectives of IEIS were met and within the agreed upon timeframe. Sure, there was a considerable shake down period afterward, but the programs and systems worked, and Oakland was able to install hardware and software, and implement the programs developed by IEIS.

There was a problem, however, in that despite the confidence of the future, promised performance improvements in PL/1 simply did not materialize, and it is just possible that some of those grade reporting programs that were running back in the late 60's and early 70's are still running on Oakland's System 360/Model 50 that is long gone.

Due to Oakland's need for a product that would adequately serve the data processing needs of its local school district constituents, Oakland proceeded to modify IEIS programs, which were in the public domain, and further to correct bugs and put in the necessary enhancements to get the job done, that job being that of meeting the needs of its local school district constituents. By the way, Oakland also found it necessary to rewrite all of the programs in COBOL, simply so that some of the jobs would eventually come to completion. There was a lot of calypso music in the Oakland data center in those days, because there were a lot of people and a lot of jobs in limbo.

Thus, although the programs developed by IEIS did work, they didn't perform satisfactorily for Oakland's needs, and Oakland merely capitalized on the efforts of its then partners, namely Macomb ISD, Wayne County ISD, and City of Detroit Schools, and it developed a unique system that met its needs and the needs of its users.

There was a period of time, from 1969 through 1973, that Oakland was strictly going it alone. Macomb ISD basically stayed with the hardward and software with which it had been operating, as was the case with the City of Detroit Schools. Wayne County ISD continued doing just as much as it possibly could with its 029 keypunch, which was the extent of its data processing equipment at the time. Their programming consisted of multi-function control cards with star wheels.

In 1973 Wayne County ISD recognized the need to either increase the capacity of its 029 keypunch or to jump into data processing full board, and Wayne went from the 029 to a System 370 Model 158, and with Wayne's normal calm optimism which persists even until this day, they merely intended to install this 158 and go full boar overnight.

Again the excellent working relationship between Clare Ebersole and Rex Wood came into play, and Wayne agreed to enter into a partnership with Oakland Schools providing Wayne with the use of the programs Oakland had developed based on the IEIS product. The system which the programs comprised was now called RAMS, standing for Remotely Accessible Management System.

I want to again emphasize that the name RAMS was coined by somebody else long before I came on the scene. I take no credit or blame whatsoever for the term, whatever your perspective, and although I have suggested a name change a couple of times, I do believe that overall the name RAMS, Remotely Accessible Management System, is an appropriate description of our product.

As a part of the partnership entered into by Oakland Schools and Wayne County ISD in late 1973, it was agreed that Wayne County ISD would get full use of and partial ownership of the programs that then comprised RAMS, and it was further agreed that Oakland and Wayne would pool their resources and form a single development staff to meet their advanced development needs jointly.

This is basically the history of RAMS, up to the time of the formation of the RAMS Cooperative, from one persons perspective, mine.

## THE OPERATION OF THE RAMS COOPERATIVE

The RAMS Cooperative has now been in operation some 6 1/2 years. We have had many problems, and a few successes, but I believe that rather than to bore you with descriptions of many of the problems and waste a great deal of time, and rather than to terminate this sessions embarassingly early by commenting on our successes, I believe it best to merely describe to you how the RAMS Cooperative operates today.

I mentioned the difficulty I had trying to administer a staff of 14 members operating under 3 different personnel policies, namely Oaklands', Wayne's, and the RAMS Cooperatives, and I am happy to say that operations have been considerably better since we finally developed a single set of personnel policies for the RAMS Cooperative and successfully got these policies approved (in 1978).

Certainly another problem that was significant for quite some time was that of trying to get two separate strong willed agencies to agree on anything. The words "give" and "take" were stricken from the vocabulary, and many of the negotiations consisted of arms folded and a shrug with no budge. And, of course, during all of these early negotiations the RAMS Cooperative was in the middle, and we always felt that we were considered at fault when we could not get the two partners to agree on areas where they simply would not or could not.

It finally became apparent, and it was no easy solution because it took considerable discussion and further negotiation, that in cases where one partner wanted a capability that the other one didn't want, or where one totally disagreed with the others requirement, we merely collectively came up with the idea of using options and of adequately protecting the partner who did not want the specific option from getting trapped into it. Thus one finds that RAMS is extremely flexible, full of options, and thus RAMS is able to meet the specific legal requirements of the Federal and State Governments, yet provide considerable flexibility to the local school district user.

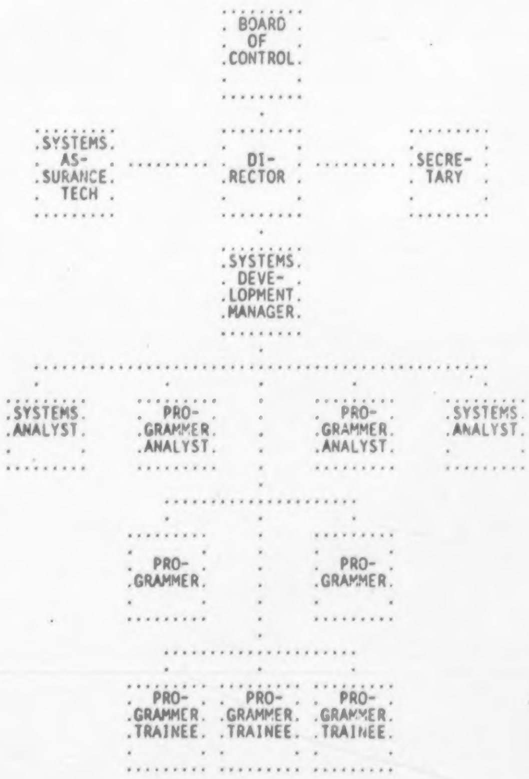
Quite often in presentations we are asked, "Can you do thus and so?", and the response is usually, Yes, depending upon which options you select. Then we get the question of "Who should be responsible for data input, the Host ISD, the district administrators, or the school principal? What do we recommend? Again our answer is "How you want to run your business is your responsibility and not ours. We have merely tried to provide the tools and the flexibilities so that you can run your district the way you want to, yet have the capability of a powerful management system at your fingertips."

## RAMS COOPERATIVE ORGANIZATION

The policies of the RAMS Cooperative are set by the Board of Control, and administered by the Executive Director. The Board of Control is comprised of the Superintendents of the member intermediate school districts, each of whom has a weighted vote in proportion to his contribution to the RAMS Cooperative budget. The Board of Control generally meet quarterly to review policies and progress, and to establish project priorities. An Executive Committee consisting of the Chairman and Vice-Chairman has been empowered by the Board to take necessary action as needed between board meetings.

The current RAMS Cooperative staff organization is presented in Exhibit A:

EXHIBIT A
RAMS COOPERATIVE STAFFING FOR 1980-81
(13 STAFF POSITIONS)



A thorough set of job descriptions and operational policies have been reviewed and approved by the Board of Control.

New project requests, programming enhancements, or program bugs are referred by local school district personnel to the Host ISD. If the problem situation is a program bug that needs to be fixed almost immediately, the Host ISD fully understands that it has the responsibility of making the correction to satisfy its local district. That Host ISD also has the responsibility of releasing the change so that the other ISD Participants can be aware of and avoid the same problem.

If the request is for a new product or program enhancement, the Host ISD staff member will complete a Programming Request Form (see Exhibit B), obtain the signature of his or her Data Center Director, and forward it to the RAMS Cooperative.

Upon receipt of the request by the RAMS Cooperative, we notify the other ISD Participants and request their concurrence. If we do not get concurrence from the majority of the Participants, we have the option of merely referring the request back to the requesting Host ISD, with the indication that the request was not mutually acceptable to the other Partners. That ISD can then make the change itself, with the understanding that it is fully responsible for this "Local Fix".

As an alternative, the RAMS Cooperative will sometimes indicate that we will proceed to make the change, anyway, and merely put it in as an option to benefit the requesting ISD Participant, with adequate safeguards to protect those who do not want to use this option. Obviously, whether we do a program change for only one partner or not depends upon our job load at the time, and the timeliness within which the requesting partner expects a response.

Often, in fact most of the time, the requests are of a nature unfamiliar to us. One must realize that RAMS Cooperative staff members have not worked in local districts or in ISD data centers, and thus we are not specifically familiar with many of the current procedures of student accounting or school finance accounting. Thus we find it necessary to ask many questions of professional and data processing staff members of the ISD Participants, and our designs are really their designs; we are merely the facilitators.

After we jointly develop and agree upon project and program objectives, the data base layouts, field descriptions, project flow, etc., RAMS Cooperative Systems Analysts develop detailed program specification. The detailed program specifications are submitted to the professional and data center staff members for their review and a subsequent approval. When finally approved, and often this is a lengthy process, the program specifications are handed to a RAMS Cooperative programmer with the assignment of coding, testing, and debugging a program that will exactly meet these specifications.

The format and contents of program specifications have been standardized and all programming is done within standards set forth in the RAMS Cooperative Standards and Procedure Manual. All code must be structured, self documented via adequate comments throughout the program, thoroughly tested, and fully documented so as to be clearly understood by local school district users via User Documentation.

When the program and documentation are completed, the source and documentation are released to the data centers for further testing and implementation.

The process isn't perfect. The whole matter is considerably complicated by the fact that there are large numbers of people and large numbers of institutions involved, and the increased interactions of these numbers simply greatly increases the communications difficulties and complications.

Our primary objective over the past 5 years has been conversion of the RAMS programs developed by Oakland Schools in the early 70's and based upon the IEIS concepts to a data base management system utilizing IBM's IMS VS software. To date we have delivered the IMS Test Processing Systems and Student Services System, and we are close with a Payroll/Personnel and Finance System.

We are making progress, we are working together, and the name RAMS Cooperative is still apropos.

EXHIBIT B

# RAMS COOPERATIVE

## PROGRAMMING REQUEST FORM

CONSULTANT

1. DESCRIPTION OF REQUEST

2. REASON(S) FOR REQUEST

| | | | |
|---|---|---|---|
| 3. REQUESTING ISD | 4. REQUEST NO. | 5. REQUEST TYPE: ☐ NEW PRODUCT<br>☐ ENHANCEMENT<br>☐ MAINTENANCE | 6. DATE NEEDED |

| | | |
|---|---|---|
| 7. CONSULTANT APPROVAL (SIGNATURE) | 8. DATA CENTER DIRECTOR APPROVAL | 9. DATE |

RAMS COOPERATIVE

| | |
|---|---|
| PROJECT NAME | PROGRAM # |
| APPLICATION NAME | PROJECT # |

PRELIMINARY RESPONSE:

| | | |
|---|---|---|
| RAMS COOPERATIVE ACKNOWLEDGEMENT (SIGNATURE) | PROBLEM NO. | DATE |

# APPLICATION SOFTWARE: BUYER'S GUIDELINES

Robert M. Delf
Vice-President, Marketing
Columbia Computing Services, Inc.
8611 South 212th St.
Kent, WA 98031

## Application Software: Buyer's Guidelines

This paper is written under the assumption that the user has decided to explore the possibility of purchasing software, if possible, rather than doing in-house development. The following general topics are discussed:

### I. Defining Requirements

- how to develop an RFP for software
- where to go to solicit bids

### II. Software Evaluation

This section deals with the complex problem of how to evaluate a software product. The following topics are discussed:

- documentation
- input
- output
- flexibility
- maintenance
- hardware requirements
- training
- installation/conversion
- vendor credibility
- on-line (interactive) considerations
- benchmark
- cost
- evaluation techniques

I believe the paper will be of benefit to anyone using computer hardware for virtually any purpose, though most interest should be from the larger mainframe installations. There is no question in my mind that this topic is one of the most crucial in the data processing industry. As labor costs go up it is becoming increasingly difficult to develop software in-house. I think it is imperative that DP professionals learn how to evaluate software for purchase or rental and compare that with in-house development.

## I. Defining Requirements

### A. How to Develop a Request For Proposal (RFP) for Software

It is difficult, if not impossible for the user to precisely define requirements of a software product. If you recognize this fact early, the process will go more smoothly. For discussion purposes, virtually all software packages or "systems" can be characterized as follows:

1. data elements stored in the "data base"
2. input
3. output
4. calculations
5. special processes

The purpose of the RFP is to describe, in general terms, the requirements of each of these areas. The more detailed the RFP, the more difficult it is for a vendor to reply. However, you must include enough detail to discourage unqualified responses.

### 1. Data elements stored in the "data base"

It is useful to provide a complete list, if possible, of the data elements you expect the system to store. For example if the system stores data about students, you might want:

. student surname
. student first name, middle name
. street address
. city, state, zip code
. telephone with area code
. grade level
. sex
. birthdate
. ethnic code
. mother's name
. father's name
. emergency contact and telephone
. etc.

Include instructions for the vendor to indicate whether their system handles that data element and what the length of the field is. Having the length of the field will help you in your evaluation - don't assume that the existence of the data element is sufficient. The best software products have a certain amount of flexibility in the data elements that can be stored - find out if there is a facility to allow the user to define data elements. This facility gives you flexibility as your needs change and new data elements are required.

If it is not immediately obvious what is the meaning of a particular data element, include a short description:

"program code": indicates what type of program the student is taking
eg. 'VOC' vocational
'COL' college
'SPED' special ed.

### 2. Input

This section should indicate how data is to be input to the system, if you have a preference. Traditional systems have only operated in a "batch" environment while most systems being developed today operate in an "on-line" or "interactive" environment. As a general rule, it is desirable to have both capabilities if large amounts of data are being handled.

Batch facilities allow the user to process large amounts of data all at one time. The data may be input using a keypunch or other key entry system, optical scanning or mark sense methods, or an "on-line" terminal that collects data to be input to a batch run.

Interactive facilities normally allow users to enter, update, and display data on their own terminal - usually a video screen (CRT), possibly with an attached printer. The obvious advantage of interactive processing is the elimination of all "middlemen" - users are doing their own file maintenance. There are many potential disadvantages to interactive processing - I will deal with these in the section on evaluation.

### 3. Output

From the user's standpoint this is the most crucial area - does the system produce the types of output reports I need in the format I require?

In this section I would suggest you begin by listing all the output reports that are essential. Again, if it is not obvious from the title what the report should contain, include a brief description. In some cases it may be necessary to list exactly what the report should contain. If the system is relatively small, this list may be sufficient. If not, you may wish to expand the basic list with other output requirements. If your reporting requirements are changeable (99.9% are...) it is highly desirable to have a variable reporting facility or "Report Writer". This is a tool to allow non-technical users to develop custom reports without involving programming staff.

You must also realize that output consists of "housekeeping" reports such as edit lists, update statistics, and summaries. These are often overlooked but are very important to all but the smallest of systems. Such reports are often the only available yardstick of "how things went" during a batch update run. This requirement for feedback is also important during interactive updating where a warning or error message must be presented directly to the user.

### 4. Calculations

This section is often omitted altogether - a very serious oversight. It should contain explanations of any calculation that is not "obvious": for example, you can assume that the system performs mathematical functions correctly but suppose the system is required to calculate someone's age based on their birthdate. Should age be calculated as of January 1st? December 31st? The date the report is requested? a date specified by the user every time he requests the report?

This is a simple example but I think it serves to make the point: Don't assume anything about an item that is to be calculated by the system, unless there is absolutely no room for interpretation. Other examples of calculations:

- a student's cumulative grade point average
- a student's cumulative credits earned
- average daily membership
- average daily attendance
- proposed budget calculations
- payroll deductions

Just in the area of student accounting, payroll, and personnel there are hundreds of examples of calculations - the more "hidden" the calculation, the more important it is to find out exactly how it is done.

### 5. Special Processes

I consider these to be "complicated calculations". These are an integral part of many systems but are usually overlooked as well. Examples:

- how does the system produce a cheque? (i.e. exactly what steps does it follow and what calculations are done?).
- how does the student scheduling system actually schedule the students?
- in the payroll system, what exactly does the "year-end" process do?
- in the student information system, what exactly does the "year-end" or "term-end" process do?
- what happens to a student's record when he withdraws from school? when he graduates?

As with calculations, special processes are often hidden. If you have special requirements you must either address them here or when you do your evaluation.

### B. Where to go to solicit bids

You will notice that hardware requirements have not yet been addressed. Whether or not to include hardware specifications depends entirely on your existing situation. If you have no hardware now then the answer is definitely no! Evaluate all software possibilities first then tackle the hardware.

If you currently have hardware, ask yourself the following questions:

Is the application you are looking for significant enough to allow a change in hardware?
Can I afford to change hardware?
Can I afford not to change hardware?
Is the application significant enough (or small enough) to warrant its own hardware?

Obviously if you insist that the software run "as is" on your hardware, your search will be more difficult and perhaps impossible.

There are several sources to explore when searching for software:

1. Hire a consultant who specializes in hardware and software acquisitions to do the search for you. Such a person can also provide valuable assistance in preparing the RFP.
2. For applications in education, the major national conventions (eg. AASA, ASBO, NASSP, AEDS) usually have representation from various hardware and software vendors.
3. The Datapro Research Corporation has a directory of software products for virtually all major applications.
   Datapro Research Corporation
   1805 Underwood Boulevard
   Delran, New Jersey 08075
   (609)764-0100
4. Advertise in a major computer industry publication in the "Software Wanted" section (e.g. Datamation, Computerworld).

If it is a major application, my advice is to hire a consultant to assist in all phases of the acquisition. They are more adept at finding pitfalls and can provide a more thorough evaluation of available products.

## II. Software Evaluation

Once you have received responses to your RFP, the difficult problem of evaluation still lies ahead. In order to keep the evaluation time period reasonable, it is desirable to quickly reduce the potential vendor list as far as possible without in-depth evaluation - hopefully down to 2 or 3 names. This can often be done by taking your "absolutely required" list and eliminating vendors who can't satisfy all items on the list. Throughout the evaluation process, it is useful to keep a "good points/bad points" list for each vendor with entries on the list for each category outlined below.

### 1. DOCUMENTATION

This is not listed first by accident - the documentation is the single most important feature to examine. If you are not buying the source code (see Maintenance section) then the documentation is your only link as to the function of a program. For very small applications, documentation may consist only of operating instructions (including error messages and corrective action required). Many applications written for micro computers are totally interactive and have virtually no documentation. Be wary of these products unless you are willing to "write-off" your investment. Without documentation you must continually make assumptions as to what the program will do in a given situation. If your assumptions are always correct then you will be fine. But what if they aren't correct? And what about other peoples assumptions?

Documentation is not to be confused with sales literature. That may sound obvious but I have seen situations where they are one and the same. Good sales literature should list features of the system; good documentation should describe all aspects of the system in detail.

a. Operating Instructions - Batch Runs

- how to set up each type of run
- explanation of parameters
- narrative for each transaction explaining how the system processes it
- narrative for each output report explaining options, calculations, etc.
- dictionary of error messages that appear on edit lists including probable cause(s) and suggested corrective action.

b. Manuals for the End User

- non-technical explanation of how to use the system
- terminal operator's guide for interactive systems

c. System Reference Manuals (if applicable)

- explanation of calculations, special processes
- record layouts
- file layouts

d. Programmers Guide (if applicable)

- technical descriptions of the programs
- explanation of how to add new reports, transactions, etc.

## 2. INPUT

First you need to check your list of required data elements against the list offered by the vendor. When doing so, pay attention to the length and characteristics of each data field to make sure there are no serious discrepancies. If you require data elements not offered by the vendor, find out what can be done, if anything, to remedy the situation. Is there a facility for the user to define certain data elements for "custom" use?

a. Batch Input

- Are the transaction layouts logical?
- Is it easy to input the data?
- Is the system forgiving? i.e. is it easy to fix errors in input data?

b. Interactive Input

- see section 10 below

## 3. OUTPUT

If the system produces output reports (other than just interactive screen formats) then it is important to study samples - after all, these reports will be the only contact many people will have with the system.

a. Check your list of required reports against those of the vendor - ask for samples of all these reports.

b. Are the report formats acceptable?

c. Are the report options sufficient?

d. Are there edit lists for all incoming batch transactions? Are there file edit lists?

e. Is there a report generator to allow "custom" reports?

## 4. FLEXIBILITY

As your requirements change, you need to know how easy it will be to modify the software, regardless of who performs the modifications. What you are trying to evaluate here is the amount of foresight that went into the design of the system.

a. What is required to add a new report? a new screen? a new transaction?

b. Can the files be easily modified or extended to accept new data elements?

c. How much of the system is "table-driven"? i.e. to what extent can the user define the environment within which the system operates? The more the system is table-driven the easier it will be to accomodate change.

## 5. MAINTENANCE

For almost any size of software package, a good maintenance agreement can be as valuable as the original system. A crucial decision to be made is whether or not you should receive the source code for the system. Unfortunately, most purchasers feel it is to their advantage to have the source code but this is not always the case. It is virtually impossible for the vendor to honor a maintenance agreement if the purchaser is allowed to modify existing code. If a vendor claims to honor a maintenance ageeement under these conditions, read the agreement carefully to see exactly what you are buying.

a. How long is the initial warranty period?

b. How often are new releases announced?

c. What enhancements are currently planned?

d. What do you do if the program abends?

e. Is the documentation updated with new releases?

f. What is the cost of the maintenance?

g. Who do you call if you have questions?

## 6. HARDWARE REQUIREMENTS

If the application is relatively small and you currently have hardware then you probably will insist that the system run on existing hardware. If it is a "stand-alone" application, you may wish to explore the use of a microcomputer. In general you should be as flexible as possible when considering hardware requirements. If the application is large and your current computer system is near capacity, you may wish to consider new hardware for the new application. This has the advantage that large, diverse applications won't contend for machine resources, especially if you are doing interactive processing.

a. Are the required hardware resources reasonable? possible?

b. Can the system be converted to run on your hardware? If so at what cost? If it is converted, is a maintenance agreement still available?

c. Is it feasible to buy new hardware?

d. Is the system (hardware) easily expandable?

## 7. TRAINING

Again, for large systems training is a crucial part of your evaluation. You should research this area carefully because inadequate training can cause long-term problems. There should be training sessions available for all levels of user-technical and non-technical.

a. What training is available?

b. Are there written materials as well as verbal or visual?

c. How long are the training sessions?

d. Who conducts them? (i.e. what are his/her qualifications)

e. What is the cost of the training?

f. Is there any follow-up training available?

## 8. INSTALLATION/CONVERSION

At the time of installation, there are several potential problems. If source code is being converted to run on your machine, you are risking new program bugs. If there are existing data files, you must decide whether they should be converted or recreated.

a. Who does the installation and how long does it take?

b. If data files are to be converted, how will that be handled?

c. At what point do you "accept" the system as being installed and working?

d. Does conversion of existing source code affect the maintenance agreement?

## 9. VENDOR CREDIBILITY

Buying software is no different than buying any other commodity - you need to know you are dealing with a reputable firm.

a. How long has the company been in business?

b. What other products do they have?

c. What percentage of their revenue comes from this software package?

d. How many other users are there? (Get a list and contact several)

e. How long has this software package been in existence?

## 10. ON-LINE (INTERACTIVE) CONSIDERATIONS

If you are looking at a system that supports interactive terminals, then you should involve "end-users" in this part of the evaluation, if possible. You can accomplish this by using a dial-up facility, if available, or by sending a representative to an existing installation. If possible, have your representative read the "Terminal Operator's Guide" ahead of time to become familiar with the operation of the terminal.

a. Are the screen formats easy to read and understand?

b. Is it easy (possible?) to access screens at various levels without always returning to a "menu"? This is a desirable feature for the user. As he/she becomes familiar with the commands it is annoying and time-consuming to be forced to follow a particular "path".

c. What is the average response time? This is difficult to measure since it depends on the computer system as well as the software. See if you can get an answer from other users - be sure to ask the terminal operator as well as the data processing manager and compare answers (they may be quite different).

d. Calculate the "Frustration Index":

$$FI = R^2 + M + C + U$$

where $R$= average response time (in seconds)
$M$= ease of memorizing commands (scale of 0-9, 0 easiest)
$C$= cluttered index (scale of 0-9, 0 is uncluttered screen, 9 is cluttered screen)
$U$= user involvement index (scale of 0-9, 0 if you can access any screen from any other screen, 9 if you must always return to a "menu").

Rating

| | |
|---|---|
| $0 < FI \le 10$ | excellent |
| $10 < FI \le 30$ | good |
| $30 < FI \le 50$ | fair |
| $50 < FI \le 100$ | poor |
| $100 < FI$ | unacceptable? |

## 11. BENCHMARK

The easiest way to get a benchmark is to get run times from another user and convert them, if necessary. If you wish to do a benchmark on your own machine, be prepared to pay related costs and to sign a conditional contract. If possible, get run times for various sizes of runs, large and small and also try to determine resource usage for interactive terminals.

## 12. COST

There are various agreements available for most software packages - including purchase, lease, rental, and royalty. Examine your options carefully and determine all costs involved:

a. installation
b. training
c. license fees (purchase, lease, rental, royalty)
d. documentation
e. maintenance
f. modifications
g. Are expenses of vendor employees charged?

## 13. EVALUATION TECHNIQUES

a. Keep track of good and bad features of each system being evaluated.

b. Ask specific questions about how the system does something then see if the answer can be verified in the documentation (the acid test!).

Evaluating and purchasing a software product can be expensive and time-consuming. However, these costs are normally just a fraction of in-house development costs when all costs are considered. Even if the user is unsuccessful and does not find an appropriate software package, the evaluation process will produce a document outlining the software requirements. This document can then be used as a starting point for in-house development.

I think it is time everyone realized the inadequacies of the traditional data processing environment where a small group of people (often only one) "know" how to run everything. This approach causes terrible problems when the person is on vacation or sick or, perish the thought, quits their job! Well-documented software and procedures can go a long way in alleviating these problems.

# THE ROLE OF MICROCOMPUTERS IN COMMUNITY COLLEGE BUSINESS AND SECRETARIAL SCIENCE PROGRAMS

Donald A. Dellow, Lawrence H. Poole,
Forrest Clark
North Country Community College
20 Winona Avenue
Saranac Lake, New York 12983

This presentation will report on the experience of a community college in establishing a laboratory to incorporate computer literacy in all business and secretarial science programs offered by the college. The computer literacy program emphasizes microcomputer applications and hands-on experience with relevant software packages. The program is designed to provide students going into the workforce with knowledge and skills appropriate to their entry-level positions in turnkey operations. The potential for interdisciplinary simulations and future application with microcomputers will be discussed.

It is important to acknowledge at the outset that the authors of this paper do not have data processing or computer related backgrounds. At the same time, we have become increasingly aware that the application of computer technology is occurring at a startling rate and that in the near future computer literacy will be as important as our basic communication skills. Thus, we were convinced that we must begin to introduce the concept of computer literacy into most of the college's curriculums.

North Country Community College is a very small, relatively new college; and, therefore, we did not have adequate resources to computerize our own operations. We were faced with trying to decide how we could expose our students to computers without extensive investment in equipment. The availability of microcomputers seemed one logical way to accomplish our objectives given our available funding opportunities.

In attempting to address this situation, there have been several interesting experiences. First of all, we have been hard pressed to find adequate information about how microcomputers are being used in vocational programs. This has convinced us that we are not alone in our plight. The publications of AEDS have been among the best in bringing us along, but we need to be sharing more information on this subject so we don't have to re-invent the wheel so many times.

A second thing that has happened to us in trying to get into the integration of microcomputers into the business and secretarial science curricula is that we have gotten the impression that the traditional data processing and computer science people are not sure they want us to have our own computers.

Further, in seeking information about what other community colleges have been doing with microcomputers, it has become evident that having "experts" is not always a positive thing for areas like business and the secretarial sciences. It appears that often the data processing and computer science instructors encourage, either consciously or unconsiously, the "mystique" of computers. Many faculty have been intimidated by the "extensive" mathematics backgrounds that are considered necessary, by some, to fully understand computers and programming. It must be acknowledged, however, that those of us in other academic areas have probably been too willing to accept this perspective.

The first computer on our campus was received last June and that was a single microcomputer. Although we still do not have time-sharing or a minicomputer on campus, our inventory of microcomputers has increased to a total of seventeen. Much to our surprise this situation has had both advantages and disadvantages with respect to our current strategy for providing computer literacy education for our students.

The disadvantages inherent in our situation would probably be evident to everyone. Perhaps the obvious difficulty is that we have no "seasoned" full-time faculty members who can give us the technical support for complex programming and assistance in the inservice training of faculty members. In addition, because most of the faculty finished their graduate work more than five years ago when there was little emphasis on computer applications in most academic areas, there is an apprehension on the part of faculty about not being computer literate. Working out the logistics for integrating computers into the curriculm and insuring effective usage of the microcomputer lab has become a challenging experience. And finally, we realize that there are problems we are going to encounter that we don't even know about yet!

The disadvantages of being neophytes in the computer area also represent advantages in other ways. With the advent of microcomputers, many of the old ways are proving to be outmoded. In this sense, we at North Country Community College have had the freedom and responsibility of charting our own course of action. It is an exciting challenge for everyone on campus to admit to a neophyte status and work toward becoming computer literate. Although the microcomputer lab on campus was created for and is now managed by the business and secretarial science area, it is hoped that the emphasis on the nontechnical understanding of the equipment will prove enticing to other academic and vocational areas.

## THE VOCATIONAL FUNCTION OF THE COMMUNITY COLLEGE

The community colleges in this country have historically held an important place in the training of people to enter the job market directly, often referred to as vocational or occupational education. That perspective is even more true today than it was in the past. While many colleges and universities were declining in enrollments this past year, community colleges were having record-breaking years. Certainly some students were cutting the costs of the first two years of college by staying close to home; however, a significant portion of that increase in enrollment was the result of greater numbers of students entering vocational programs they perceived as bettering their chances of getting a good job at the end of one or two years of education. This windfall of students is doing great things for the community colleges, but it also puts the onus on vocational programs to stay current and reflect the needs of the "hiring communities", be they local, state, or national. The advent of the microcomputer is creating a situation in the business community that cannot be ignored.

## THE MARKETING OF THE MICRO'S

In a little over two years, the marketing strategies of personal computer companies like Radio Shack, Apple, and Commodore Pet (just to name a few) have convinced the small business community that the microcomputer is adequate and appropriate for their computer needs. These companies have sold the concept that one did not need to know a great deal of programming to use a personal computer. Instead, pre-written software would be provided which would meet the various needs of the businessperson. Although some early software was less than desireable and many of the packages today need some improvements, the concept has caught on to the extent that some powerful software packages (i.e., Visicalc) are reputed to be the rationale for people purchasing the microcomputer to run it! In addition, the concept appears to be spreading into corporations having massive data and/or information processing departments. In an article in Information Systems News (February 23, 1981), Stephen Gray reports that a considerable number of Fortune 500 companies have found their executives and departments using microcomputers to process the information they needed quickly and couldn't get from their own data processing departments.

Although there are skeptics who still question whether microcomputers are the answer for the small business (i.e., Doring, 1980), the proof seems to be in the sales figures of the major competitors and the apparent scramble of companies to get into the market. At the community college level, we see the impact of the micro's in the business area in the form of businesspersons asking for information on how, where, and when to purchase their own small business computer. We also hear the implied requests about the skills of our graduates. Will our business and secretarial science graduates feel comfortable in keeping books, managing inventories, and doing the dozens of other things the small computer is capable of doing? These are tough questions for those of us who could hardly spell "microcomputer" a year ago.

## MEETING THE NEED: A MICROCOMPUTER LAB WHERE THE COMPUTER LITERACY FOCUS IS ON APPLICATIONS

In 1980, North Country Community College applied for and received funding for a Vocational Education Grant to equip a microcomputer lab that would emphasize the business and secretarial applications of the equipment. As a result, a laboratory was equipped with fifteen Commodore Pet microcomputers and a considerable amount of business and word processing software packages. An additional Apple microcomputer was placed in the Learning Resource Center to complement the one existing Apple used for faculty and administration training in computer literacy. The equipment configuration included twelve 40-column screen terminals, three 80-column screen terminals, five dual disk drives, 2 dot-matrix printers, and one letter-quality printer. Six of the machines are tied together in a network system and the other nine operate independently with disk or tape systems. The equipment was delivered and installed in January of 1981.

In the VEA Grant we made several assumptions:

a) all career-oriented business and secretarial science students should be computer literate, to the extent that they were familiar with how microcomputers were being used in their occupational field, and

b) they would have enough knowledge of microcomputer equipment to generalize basic concepts from one microcomputer brand to another, and

c) they would acquire enough skills of data entry to be able to use with minimum skill levels the equipment itself, and

d) have enough knowledge of programming to understand how canned programs are designed, and

e) acquire knowledge about and skills in using software packages appropriate to their occupational field (including where to purchase, what makes for "good" and "poor" software, and appreciation for certain utility programs like Visicalc.), and finally

f) it is important for students to understand the rapid merging of data processing and word processing activities and functions into a new field called "information processing."

One of the things that is conspicuously absent from the above list is the ability to program microcomputers. Although we would not deny many people need to know how programs are written, there is a serious question about how necessary the ability to program will be to the employer. Herein lies the rub. Those who have been in the computer field for any length of time seem to have considerable difficulty with this concept. It is assumed that the small business person would not find software packages that would do exactly as he/she wanted and would want to have programs written. Has anyone ever found the perfect microcomputer or the perfect car, how many modifications do we make in those pieces of equipment? Like everything else in life, we have to adjust our wants with the status of our pocketbooks. Programming a package for a business should be done by someone with considerably more programming that one course or two in BASIC. The average small business employer cannot afford to hire a worker at that rate, other than on a consulting basis. It, therefore, appears to be most logical to prepare our entering students to be the "users" of computers; most probably like they are the users of cars, typewriters, televisions, and countless other pieces of technology. In the November, 1979, AEDS Moniter, John D'Angelo introduced the concept of the microcomputer as a "pencil." The implication for us was that microcomputers, like so many other inventions (i.e., the pencil), are to be used to do something and that should be taken into consideration when working them into the curriculum.

### A PROGRESS REPORT

In the current semester and next fall, the instructors in the Business and Secretarial Science programs will be field testing activities in the microcomputer lab with their students. An accounting instructor has a sophisticated general ledger accounting system that he is working into his accounting classes. The instructor will design small businesses for whom the students will keep the books. The general ledger package is typical of commercially available software packages for mainframes, minicomputers and microcomputers. Some of the students are in our Intro to Data Processing course where they will learn the basics of the microcomputers. As an experimental situation, many of the other students who are not in the data processing course will learn the basics of the microcomputers from short learning modules or through peer teaching.

The accounting instructor does not have a computer background and is learning with the students. So far, the instructor's lack of computer background has been an advantage; knowing how frustrating it has been learning software packages has brought him closer to the students who are going through the same situation.

In the Secretarial Science curricula, there are students enrolled in one and two year programs. Both will be involved in learning modules that require them to learn how to use the microcomputer with a commercially available software package. One of our objectives for the program in Secretarial Sciences is to prepare our students for the new status for secretaries when "information processing" is the norm rather than the exception. Because many of our graduates go into small business situations they will immediately be faced with a need to feel comfortable with using a small business computer for the word processing and information or data processing functions. The secretarial science graduate who has good typing skills and a knowledge of the functions of a microcomputer will be in a better position to command a higher salary and have greater access to management. We do not have the answers yet about how best to integrate the word processing on the microcomputers into the curriculum. This will be our objective for this first year of the microcomputer implementation.

As a way of carrying our assumption one step further, we are planning to create and equip a model office on campus where there will be a complete microcomputer setup with letter-quality printer for word processing and software packages for various business applications. In addition, there will be a stand-alone word processing unit with the most current software packages. After students have had exposure and preliminary training on the microcomputers in the laboratory, they will be sequenced through the model office completing real-time work orders for various departments of the college. All two-year Retail Business Management and Secretarial Science students and one-year Secretarial Science students will be required to spend a minimum number of hours in the model office utilizing the information processing equipment. It should become obvious to them very quickly, as it has to us, that the two microprocessing units are the same and their functional capabilities are merging rapidly. Students will be keeping inventories for the bookstore, completing single and multiple copies of business communications and doing some complex text-editing. We believe this approach will give the students a realistic idea of how to meet the increasing computerization of the world of work. This will better prepare them for the job situations that are being created right now in our own community.

## TOO MUCH TOO FAST?

There are those who have advocated that educators not jump into the microcomputer craze too quickly. It is true there are many unknowns and many possible dead ends. Yet is seems this revolution that Toffler (1980) is talking about will not be denied. Those of us who are close to the vocational role of education feel we cannot wait for the necessary research. As recently reported in Time (Taylor & Moritz, 1981, p. 68), "Sales of small computers are increasing at a rate of 50% to 60% a year..." This means our clientele will want students who are computer literate and microcomputer wise.

## REFERENCES

1. D'Angelo, J. The microprocessor as pencil. AEDS Monitor, Oct/Nov/Dec, 1979, pp. 14-16.

2. Doring, P.F. The bleak future of small-business computing. Creative Computing, November 1980, pp. 52-53.

3. Gray, S.B. Some corporate MIS departments admit they find micros useful. Information Systems News, February 23, 1981, pp. 37 & 40.

4. Taylor, A. (Reported by Micahael Moritz) Small-computer shootout. Time, March 2, 1981, pp. 68-69.

5. Toffler, A. The Third Wave, New York: William Morrow & Co., Inc., 1980.

# A PARADIGM FOR COMPUTER-BASED INSTRUCTION: RENAISSANCE OF THE AFFECTIVE DOMAIN[1]

George Donoian, Ed.D.
Business Administrator
Detroit Public Schools

This paper delineates the problems facing educators who must determine whether computers will be used in a constructive, humanistic manner or merely as another mechanical fad. It offers expectations that computers have a creative function in education if the student and teacher are encouraged to use it for freedom of choice in an expanded curriculum.

As an urban educator for over twenty-five years, I have been both a witness to and a participant in the educational process. In that time, I have seen the rise and fall of many tides upon which American education has ridden. Public demand for staff accountability as the most current wave has brought many responses that include objective-referenced testing, teaching by objectives, diagnostic-prescriptive teaching and a host of trendy notions which, like other past responses (classroom TV, team teaching, ability grouping, etc.) will hold off the critics for a while.

The public will not be thwarted easily by these "program priorities". For example, the current expectation for high school competency skills presupposes that students with twelve years of attendance, and hopefully, education should graduate with minimal skills in reading, writing and math.[2]

As a direct response to this pressure, schools are emphasizing cognitive teaching, and thus hoping for cognitive learning. Indeed there is mounting evidence that this method has been working. However, this heavy stress upon cognitive skills and knowledge acquisition has been paying off at the grave neglect of students' affective and psychomotor development. A close look at this phenomenon is necessary.

A study of computer-based education software will convince the casual observer that the curriculum has become cognitive. What makes this situation more problematic is the degree of commitment school districts have defensively embraced the cognitive realm at almost the total absence of anything affective. I acknowledge and admit that computer course work is certainly intriguing for many youth who enjoy the novelty and mastery over new material, but allowing this heavy dosage of cognitive learning to continue and become intensified, then dare we ask our graduates about sheer "facts learning". Our students tell us startling tales about the shortcomings of their education. Without picking on computer science or mathematics teachers specifically, we are beginning to reap the effects of their overstressed exact sciences and procedures. In most cases, I have found that when achievement is not regarded as a process, but more precisely as a product with its own intrinsic value, then today's education loses its relevance and raises itself above all else. In this exalted position, cognitive learning becomes a sacred cow—an untouchable commodity requiring total commitment to the demands of the task without human sensitivity to student needs and perceptions. Is this what computers are bringing into our schools?

### Prediction of Things to Come If Trend Is Unchecked

It is my purpose to forecast that as the present trend continues in education, the stress on cognitive teaching and learning as exemplified in computer-based instruction, but not limited to it, will logically and inevitably lead our students to reject and rebel against our defensive measures to redeem ourselves as educators. By continuing our pressurized ways to get students to achieve and thereby justify our presence and positions in schools, we will pay dearly for having been so unbalanced in our curriculum. Without humanistic efforts to offset our heavy dosage of cognitive learning, tomorrow's students will be denied fundamental skills, knowledge and attitudes which could help them to become better people, especially as we look to the next decade with the burgeoning use of computers taking over human work. Depersonalization of society may be inevitable, but the schools should not be accelerating nor promoting this process by training the next generation of adults to accept, enjoy and function without affective substance. Therefore, education stands at this threshold of progress and responsibility--in what ways we respond and accomplish this mission will affect the whole fabric of American public education and the coming generations.

### Current Curriculum Problems

In the headlong pursuit of achievement we school people have nearly forgotten to ask some fundamental questions about the content and methods which have recently been introduced. The quest for cognitive mastery has not been all that "bad", but it has left some questions unanswered and created a great deal of uneasiness about the so-called success of the achievement orientation, along with the systems approach to learning. One of these pressing issues has been that of relevancy. What materials and what methods seem to produce the best results? What values do they engender? And for whom? In reaching for answers, too many of our teaching colleagues have lost sight of the students' needs, wants, wishes and interests. Thus the system is the *end* and not the *means*. Because cognitive teaching lends itself to easy description and mechanical manipulations, we have found that computer-based instructional materials, for example, lack values clarification, humanistic touches and affective thema.

An issue scholar-teachers should raise is the matter of form (*and not the function*) of all the necessary skills and concepts which include the *when* or *where* these should be introduced, reinforced and mastered as is the concern of *how do they fit*. I think education is a process of student refinement, of increasing their sophistication, of explaining the complex in understandable terms, of having students explore the unknown by converting it into known, of their own making what seems isolated into an integrated complete whole, filling in their gaps of knowledge and providing students with a success vehicle through mastery. Cognitive learning alone cannot accomplish these expectations which are so basic. Computer-based instruction has got to be more.

What is most important is that the content we are stressing should have a direct bearing upon our students' life in the future as well. Why should there not be some synergism in our curricular offering so that students can be expected to grow educationally to solve complex mathematical problems with computers? Or to expand their social studies horizons beyond their immediate neighborhoods so that they can analyze and solve global problems using computer-simulated models, for example?

### Resolution: Humanism and the Renaissance Motif for Computer-Based Instruction

While the spectre of producing a generation of cognitively trained students looms mistily, it is nevertheless a possibility which could become a reality if we allow ourselves to be carried along by a tidal wave of convenience and sentiment which embraces achievement *at all cost*. This perception is further reinforced by the egalitarian idea that every student is capable of learning and we teachers are discovering that computer-based instruction just may be the ticket allowing us to climb aboard this successful educational vehicle. In response we could answer with our progress charts and show how much we have taught our little darlings as the evidence of our work. Thus we should be able to silence our critics who want our heads with their demands for accountability.

It is my opinion that all this effort and amassed achievement data will not convince our students or their parents that even with expensive computers and intricate procedures that any fundamental education has taken place. Let me repeat: We will not convince the public that we are successful.

How do we know this is true? Humanism has not taken hold and does not seem to be driving the schools toward an inevitable place of eminence, that is, *providing an education for all to become universal beings*. Therein lies the problem - the disparity between the common man becoming uncommon through study and travel, and eventually through work and service. Merely providing transistory school achievement opportunities cannot produce lasting Renaissance qualities and values in our students who must survive and hopefully, thrive in a environment with shrinking resources, international hostilities, economic upheavals and advancing technologies. We educators have an obligation that goes beyond just being professional - we need a mission of greater importance. I refer to the Renaissance period which awakened and hastened humanity out of the Dark Ages and thereby quickened the need for personal knowledge, freedom and fulfillment of capabilities. Indeed, the process began modestly and inauspiciously with King John signing the Magna Carta on the plains of Runnymede in 1215 A.D. Surely this event had many predisposing events and many yet to come before human rights as we enjoy them became realities in these United States.

Bloodshed and property loss have been the ultimate price tags for establishing and guaranteeing the inalienable rights we cherish while other people around the world await their turn to achieve. We have earned ours by difficult means and, therefore, stand to lose some if each generation does not reaffirm these freedoms. This is accomplished by teaching history and values as we pass our political and cultural heritage onto the next generation - our students. Computers and computer-based instruction cannot do this.

Education should be at the forefront of that movement to safeguard the rights of the coming generations. We do not need another million years of struggle to discover the efficacy of a humane educational process as the means of improving mankind and establishing the wisdom to do something about it. We must act now to teach humanism as the first art in all our schools.

## The Renaissance Motif Explained

What must we do to bring about a Renaissance in our midst? How will we recognize a rebirth of this spirit in American schools? All that we propose must bear the hallmarks of a humanistic education if we are to be producing students as citizens who will be functioning effectively in a world of problems requiring answers.

For education to be humane we need much more than a flurry of activity directed at teaching students to achieve cognitively. That is simply quantitative. An achievement campaign is only the opening thrust of our Renaissance campaign with its pursuit of vast amounts of knowledge. Education will experience success in demonstrating that children can, will and do learn.

However, the Renaissance Motif calls for more. The humanist theme requires an imaginative response from us to broaden our students' horizons along with their reservoir of knowledge and skills. For students there must be increased offerings and learning experiences which will enhance their literacy, artistic and cultural lives and not at the neglect of their scientific knowledge if they are to become universal beings during this modern Renaissance era, especially as we rush them toward the Computer Age that is now upon us.

In addition to our conviction and commitment that today's youth can learn, we educators must provide for a greatly expanded curriculum which will be the basis for a humanistic education. In short, the quantitative aspect of our proposed studies will require more resources and a greater imaginative use of the ones we already have available to us which must include the computer for instruction and management.

More significant than the extent of our offerings is the matter of our methodology and philosophies from which springs the essence of humanism. What must we do to be certain that humanistic ideas and motives permeate and saturate the curriculum? How do we guarantee a humanistic education, one which promotes the development of the Renaissance Motif preparing everyone to be universal beings?

## A Paradigm for Teaching and Learning Characteristics of a Humanistic Education

In addition to the hardware of terminals and a plethora of rich learning materials, the Computer Age must be ushered in with humanistic substance if we are to witness a true rebirth of the individual and recognition of his worth. Students of the Computer Age and their schools will need to be reorganized for effective implementation of the Renaissance motif as the centerpiece of a working humanistic educational philosophy.

To be evident, the Renaissance Motif will be recognized readily by the following characteristics: First the curriculum, school, staff and society must be geared to serve the centrality of every student and his self-development by providing for his *active* participation in the design and direction of the curriculum. For as long as students are prevented from making choices and decisions about what, when and how they are to learn, then they are merely *passive* witnesses to the learning environment. Without the students' *active* involvement which would provide them with numerous opportunities to develop and practice their skills in exploration, discovery and analysis and synthesis, the so called learning is merely a mechanical process. They need to learn to think critically and acquire study skills so they can formulate their own problems, explore, discover and then arrive at their own conclusions and solutions. They need to develop the courage and desire to tackle new problems. Humane teachers will be required as well as the computers to accomplish these formidable expectations for the classroom of the near future.

Second, we must become living examples of the imperative to cherish students' purposes and perceptions as equal to our own. We must give up our long-held beliefs that we know better than they about what may be good for them to know. I believe that it has been this factor which we have conveniently termed as "lack of motivation" on their part. Suppose we turned it around and allowed our students to study the curriculum content which they themselves see fit for their purposes? We probably would witness a quantum increase in motivation and learning - a veritable revolution in achievement which has intrinsic value rather than offering them external rewards, i.e., good marks. Again the computer is perfect for this requirement. Are we ready now to implement an open-ended curriculum based upon student needs?

Just as it was for the first Renaissance, infallibility of knowledge and the purveyor must give way and be replaced by variability and tentativeness for our present-day scholars to become humanistic. Logical inquiry and defense of any issue will depend upon the skills we teach. Our students must learn that there are, at least, two sides to any question. How much more learning can take place when our students skilled in debate and argumentation can engage in lively mock courts and debate sessions for utilizing their talents? The computer can dutifully provide a socratic questioning mode to teach debating skills to our inquiring students, as well as connect two students hundreds of miles apart to discourse via the video display or challenge each to solve difficult math problems.

Next, our humanism must reach beyond the narrow confines of our own time parameters. Humanism by definition embraces a time management which maintains the student's own purposes and perceptions at the very core of his education, not our own directives. What we perceive as adequate learning time may not be so for our students who are the learners. Therefore, we teachers must be ready to dwell upon a unit of study longer than we had planned because the learners may be enraptured by it or feel it has not been adequately dealt with in depth, therefore requiring several more weeks of study. The computer again is a natural adjunct to the teacher and the textbook by being a patient, helping tutor for students and by providing additional resources and practice.

Fifth, in keeping with students' purposes, perceptions and time management, we ought to be certain that their personal wisdom is respected and honored, since we already know student excursions into the world of self-directed learning are the most poignant ones. Despite that knowledge, we discourage, and worse yet, penalize students for doing just that for causing us to deviate from our lesson plans and semester designs to cover a curriculum that is usually sterile, deadly and, most likely, useless.

Instead, we should be wondering how as educators dedicated to humanism we can send our students onto wide-ranging cerebral trips, which may seem like they are seeking the obscure, the trivia and useless bits of information that will provide them with significant knowledge, answers to difficult questions and solutions to their problems. Ready access to more curricular content beyond teachers and media centers' reach can best be made available to students via the computer having accessing capabilities of almost limitless dimensions. Media centers today are not adequate for students to research in depth. Therefore, the computer can "take" students to new sources of knowledge without leaving their schools.

Finally, the value of humanistic education has to be in sharing it with one's fellows. Vocationally, too, our students will measure its utility by what use they can make of it in the marketplace and the ego-enhancing qualities humanistic education bestows upon its recipients, not the least of which is the satisfaction of working and serving others, be it in the arts or sciences. Tapping into computer terminals and conversing with remote resources are capabilities we are just beginning to explore with our computers in the classroom. A truly universal learning environment can only be provided by the untiring eye of the electronic brain.

## Conclusion

I believe that education is at the crossroads today where it must decide its own future. It cannot continue to present excuses, reasons and catchy notions about how it is attempting to overcome problems. One such way out of this dilemma is to be preoccupied with teaching for the sake of achievement - only a step out of the quagmire. However, this effort is insufficient to carry education beyond its point of departure on its journey toward humanism.

I propose that education must be quickened to a point where it resembles a second Renaissance which includes a philosophic outlook embracing the following six parameters comprising a reawakening in teaching and learning:

1. That every student is involved actively as a learner,
2. That each student's learning purposes and perception be respected,
3. That knowledge is variable and tentative, not infallible and our teaching designs and students' learnings are dictated by modern scientific and social facts,
4. That the student's time management is more relevant than the teacher's own chronological limitations for becoming,
5. That student judgment and wisdom prevail when we are determining the what, how and why of learning, and finally,
6. That to students the value of a humanistic education is in its use for self-enhancement and for work as service to oneself and others employing a foundation of learning in both the arts and sciences.

We will witness a second Renaissance when our schools operate with these six characteristics of a humanistic education as the epicenter for the coming computer age. A greatly expanded curriculum will permit our students to have freedom of choice, personal dignity and increased knowledge and skills befitting universal beings who are equally at home in the arts, as well as the sciences. Our students will be modern-day Renaissance scholars ready for the twenty-first century. Only through the imaginative and humanistic use of computers as facilitating instruments of instruction can we begin to witness a Renaissance in our times.

---

[1] The author claims sole responsibility for observations and opinions herein.

[2] Schools in the State of Virginia will be issuing four different high school diplomas to graduating seniors based upon their competency testing results.

[3] See A.E.D.S. 17th Annual Conference Proceedings, "Humanism for the Computer Age in Education: Planning for Man/Machine Synchronicity", by George Donoian, pp. 321-24, 1979.

# ACADEMIC AND NON-ACADEMIC BENEFITS
# FROM USING PLATO
## A CLASSROOM TEACHER'S VIEW

Lynne Erting and Phil Mackall
Kendall Demonstration Elementary School
Kendall Green, Gallaudet College
Washington, D.C. 20002

One teacher at Kendall Demonstration Elementary School, Washington, D.C. used some PLATO lessons with her elementary-aged (9-11 years), hearing-impaired students. These students are functioning at a first to second grade written English level and are known for their disruptive behavior. This article describes how she used these lessons on the computer and some off-line activities to improve her students' academic and non-academic skills.

Kendall Demonstration Elementary School (KDES) is a federally funded institution which provides educational services for hearing-impaired children in the metropolitan Washington, D.C., area from the onset of deafness to age 15. It also serves as a national center for developing, testing, and evaluating materials, methods and programs for use in other schools for the hearing impaired.

As a part of its mandate to test and develop materials, Kendall has collaborated with Control Data Corporation (CDC) in Minneapolis, Minnesota, for two and a half years in: (1) assessing PLATO's[1] current potential for use with hearing-impaired populations; and (2) in developing guidelines for, and examples of, lessons that are specifically designed for delivery to a hearing-impaired population. Existing materials tried with our students[2] included: (1) the math part of CDC's "Basic Skills Learning System, a set of lessons originally developed as an alternate method of learning for 16 to 24 year old, functionally illiterate adults; (2) the Colorado Hearing and Speech Phase II Lessons, a set of visual discrimination, visual memory, and sequencing lessons originally developed for learning disabled children (K-3) and modified for use with hearing-impaired children; (3) various math lessons designed by several authors[3] on PLATO; and (4) a few language and social studies lessons.

Our students have also participated in a communication experiment via PLATO with hearing-impaired programs in Hawaii, Illinois, and Pennsylvania.[4] Results of these first experiences with PLATO's lessons and capabilities were positive.[5]

The developmental efforts that followed these initial experiences with PLATO resulted in a set of guidelines for writing computer programs for hearing-impaired populations[6] and a set of Animated Sentence lessons[7] which are now being tested at our site.

The focus in our earlier reports has been on the academic areas with many non-academic areas as a noticed, but basically unimportant, factor. This paper will detail one teacher's use of PLATO with a class of elementary-aged students at Kendall where the initial focus was on non-academic areas, but also included academic ones as a secondary issue.

## POPULATION DESCRIPTION

The class consists of five students; two females and three males with an age range of 9-11 years. One of the students comes from a Brazilian family where Portuguese is the mother's native language, although she speaks English to him at home. Two of the students have moderate hearing losses while the other three are profoundly deaf. All of the children benefit tremendously from the use of sign language in the classroom as well as in their daily lives. IQ results on the WISCR show that students range from 71 to 123.

This is a very energetic class. One student is on medication for a learning disability which results in hyper-activity and corresponding behavioral problems. The other four students have no other identified disabling conditions. Only the hyperactive child has been identified as a "behavior problem" per se, but one other boy is highly distractable and has a very short attention span. All of the students have frequent difficulties attending to a group lesson as well as to specific individual tasks. This distractibility plus other immature behaviors require a very structured environment.

Classes for the hearing impaired are generally not graded and the students' academic levels are generally lower than those of hearing children of the same age. The students are grouped according to their educational functioning into the most homogeneous classes possible. The students in this class are functioning on a second grade reading level, second-third grade math level and a third-fourth grade conceptual level in science and social studies.

## Academic Problem Areas

One of the major academic problem areas for a hearing-impaired child is learning the language of his/her society, in this case, English. To adequately understand this unique problem, some background information is necessary. Unlike the hearing child who learns language auditorally, the deaf child relies more on visual cues. Because there are differences in the linguistic structure between any spoken language and any sign language, the deaf child has greater difficulty learning his societal language, even in a non-oral form.[8]

The sign language used by the majority of American deaf people is called American Sign Language (ASL). It is an un-Englishlike visual language, "a linguistic system like other languages (with) its own (lexico-semantic, syntactic and expressive) rules"[9]. Deaf children learn ASL from their parents, if they happen to be deaf, or through contact with other deaf individuals either socially or at school. ASL does not, however, have a written form. Neither is it the language used by the majority of (hearing) people in this society. Therefore, the hearing-impaired student must try to gain competency in English at school. In an attempt to achieve this goal, many education programs for the hearing impaired use Signed English, a code for representing English visually by using ASL vocabulary but putting it in English word order. Although Signed English is a valuable tool, current research does not indicate that it provides the deaf child with adequate input for the development of English skills comparable to that of a normally hearing child.

One can see, then, that learning English is in itself a problem for the deaf child. Another stumbling block is the negative attitude that many deaf children have towards learning English. This can result from the child's repeated failures in attempting to read and write a language which may have very little meaning for him/her. From the educator's viewpoint, competency in English is essential for successful communication. Elementary-aged, hearing-impaired children, however, communicate quite adequately through the use of sign language and may not be able to appreciate the need for the development of competency in English. In their efforts to benefit the child by trying to provide him/her with what they feel are some essential life skills, educators may, in reality, be forcing the child to participate in a meaningless activity which, to the child, only represents failure and frustration.

The students involved in this project are no exception to this whole problem area of English language learning. The two with moderate hearing losses have better English skills and more positive attitudes than the three with profound losses. This is natural, since they can hear more of the language and achieve greater success. The three with profound losses have fewer English skills. Two of these students are very reluctant to read and even more reluctant to write, especially creatively.

The following are some written English samples taken from these students at the beginning of the school year which more fully illustrate the aforementioned problems:

Student 1 (profound hearing loss)

hain(hair) make woman
vaney(very) peattry(pretty) roll hain(hair)
finish cut you hain(hair)

Student 2 (moderate hearing loss)

I come home.
I cook hot dog for the gril(girl).
But I see the TV
My gril(girl) wsa(was) play Ball.
My thee gril(girl) wsa(was) in the bad(bed)
But thuy(they) wsa(was) play in the bad(bed)

## Non-Academic Problems

Two non-academic problem areas which had direct effect on the classroom learning environment were: (1) the student's inability to get along with each other and (2) disruptive classroom behavior. The time invested in dealing with these behaviors became so excessive that these areas, rather than academics, began to dominate the majority of time in the classroom.

Socially, at the beginning of the year, the students were at a very egocentric level of development. They were unable to share, take turns, or be polite to one another. Pouting was a typical way to deal with disappointment. If a student did not achieve success on the first attempt at even a simple task, often s/he would give up rather than try again. Some students were afraid to take risks; i.e. to try something new without first checking with the teacher for approval. Other undesirable behaviors included picking on each other, tattling, fighting, hitting and arguing.

At the same time, there were many classroom behaviors that also interfered with academic progress. One of the major problems was distractibility or short attention span. This made it difficult to work with the entire class as a group since the majority of the lesson time would be spent on maintaining the students' attention. It also made completing a given task a problem for both teacher and student.

Another area of concern was the students' inability to work independently. This could have been a result of deficient reading skills or, as previously stated, from general dependency on teacher approval. This lack of independence not only limits teaching strategies but limits the students' growth as well.

For some students, staying within the classroom area was, in itself, a difficult task. This particular class area was an open space situation which required internal control of each child rather than external walls to keep him in the classroom.

In combination, all of these behaviors became so overwhelming that conducting class became a struggle rather than an enjoyable learning experience. A system had to be devised whereby the students themselves were motivated enough to improve their behavior. Initially, this is where PLATO came into the picture.

## LESSONS/MATERIALS

Some of the students in the class described above have had previous exposure to PLATO. With one of the students, PLATO had been used as a behavior reinforcer. This student was constantly sneaking out of class (on the way to the bathroom or to the nurse or to ...) and signing onto PLATO. Turning off his signon name did not stop his attempts. It seemed appropriate to link his strong desire to use the computer with the teacher's strong desire for good behavior. Therefore, a system was designed that permitted him to use PLATO at the end of each day IF he earned a certain number of points during the day by staying in class and behaving appropriately and IF he did not try to sign on at inappropriate times. This system proved successful.

Faced now with a similar situation, i.e. behavioral problems with the same student and others in the class, PLATO was selected for a repeat performance. It was hoped that time on the computer could once again encourage positive behaviors in individual students and also in the group. The teacher's task now was to select (an) appropriate lesson(s). The lesson(s) should be fun and, if possible, educational. Since the students would be using the terminal during their Reading/Language period, it should also be connected to those subject areas. The problem: most of the Reading/language lessons in the PLATO Lesson Catalogue are too difficult for these students.

One of the projects that Kendall was trying to promote in the Elementary Department was the use of a "group notesfile"[10] that allowed students at KDES, the Pennsylvania School for the Deaf in Philadelphia, two schools in Champaign/Urbana, Illinois, and the Hawaii School for the Deaf and Blind to exchange notes. Since the reading and writing of notes are very appropriate for a Reading/Language class, an attempt was made to use the notesfile as the reinforing activity with this group. Unfortunately, it failed. The Reading/Language levels and the sophistication required for this task were too high.

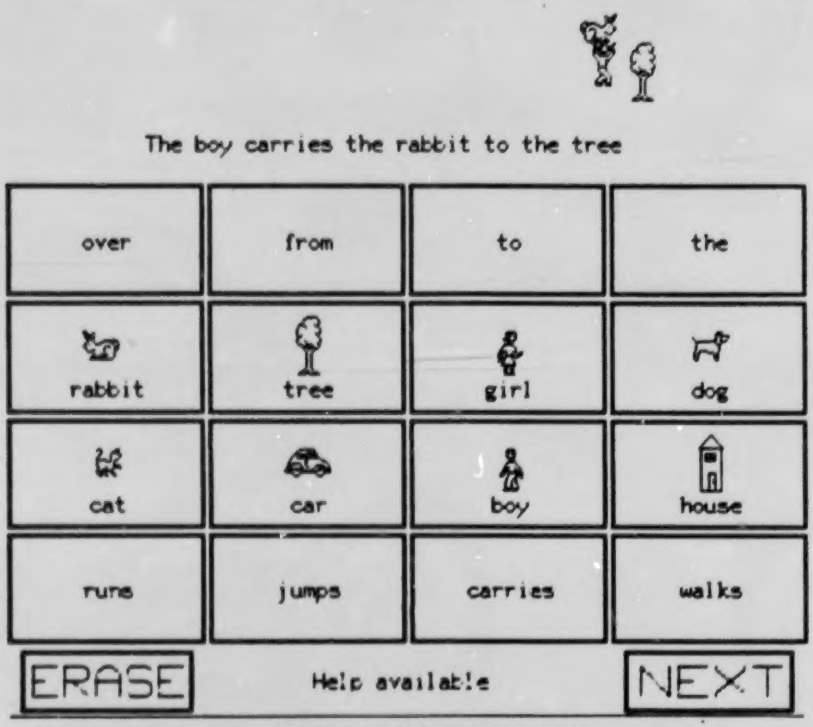
Further searches for a lesson that would be appropriate led to one that had been used with other KDES students and that seemed to be effective in increasing student vocabulary and the ability to write simple sentences. This lesson ("Osentences" or "Sentences") provides a set of boxes containing the word "the", some animate and inanimate nouns, and some transitive and intransitive verbs. Sentences are made by touching the boxes on the screen. As the boxes are touched, the words are placed above the set of boxes in the order touched. If the resulting "sentence" is syntactically correct, the student is rewarded with an animation of the sentence. See Figure 1 for an example. This lesson served as a spring board for the lessons which Kendall was developing and planning to field test and which were also used by this group of students after the lesson development effort was completed. These new lessons continue to use the idea of providing a way of experimenting with sentence construction in a meaningful format (i.e. good sentences = pictorial enactment of the sentence), but provide instructional strategies as a basis for student interaction. This was done by developing lessons that "forced" the student to make a particular type of sentence. In order to guide the student in making correct sentences, the boxes for appropriate word choices would flash (e.g. when a noun was needed, all the noun boxes flashed). Figure 2 shows an example. Since the content of the lessons was still within the domain of the Reading/Language areas, since the academic demands were much lower and since very little computer sophistication was required, these lessons seemed to fit both student and teacher needs.

To further facilitate the success of these lessons a special strategy was used. PLATO provides several different options for delivering lessons to students. One of them is their "mrouter"[11] or main router which permits three different formats: (1) a sequence which forces the student to take assigned lessons in a particular order; (2) a sequence with review which forces a student to take assigned lessons in a particular order but allows him/her to review those lessons (s)he has completed; and (3) an index which allows a student to choose which of his/her assigned lessons (s)he wishes to do at each session. The third format was used. Since the sentence lessons were considered "work", the index was set up as in Figure 3. Students were asked to make ten sentences first (work) and were then permitted to try the games (play). This format was later modified to exclude the games. Rationale for this is given later.

In the original developmental plan for the Animated Sentence lessons, a need for other types of lessons was foreseen. However, there was not enough time or money in the current project to execute these proposed lessons. Figure 4 shows a summary of the lessons that were developed. Figure 5 outlines the extra activities that were anticipated.

During the use of the newly developed lessons by this group of students, the need for two of these extra lessons surfaced. Those lessons were: (1) lessons requiring the student to create sentences for a given sentence animation; and (2) lessons with limited vocabulary for practicing the patterns learned in two or more lessons. Both of these lessons were partially addressed by off-line teacher developed activities.

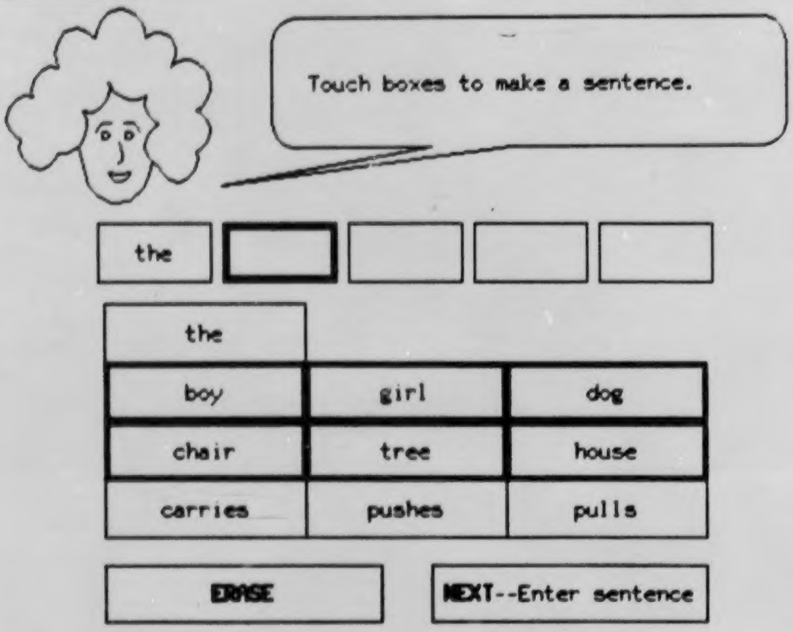
The first need was met with the aid of the Animated Sentence lesson programmer who inserted special temporary commands in the lessons that stopped the action of the animations. This permitted the making of screen copies of the anima-



The boy carries the rabbit to the tree

over
from
to
the
rabbit
tree
girl
dog
cat
car
boy
house
runs
jumps
carries
walks
ERASE
Help available
NEXT

Figure 1: Example of the "Sentences" lesson format



Touch boxes to make a sentence.

the
the
boy
girl
dog
chair
tree
house
carries
pushes
pulls
ERASE
NEXT--Enter sentence

Figure 2: Example of the Flashing Boxes in the Animated Sentences Lesson

- Lessons for Ms. Mynne E's Class

Do "a" first. Make 10 Sentences

a. Make a Sentence

Fun Lessons

b. Concentration

c. Speedway

d. 15 Puzzle

e. Pretty Pictures

Figure 3: Index Module for students

Lessons That Were Developed

1. Basic Sentences (examples below)
   a. The girl runs.
   b. The boy pushes the tree.
2. Sentences With Conjunction "and"
   a. The girl and the monkey jump.
   b. The girl and the boy carry the chair.
   c. The monkey pulls the TV and the chair.
   d. The bird runs and the cat jumps.
3. Sentences Using Prepositional Phrases
   a. The girl runs to the bird.
   b. The boy pushes the ball over the house.
4. Summary lessons that accept multiple formats.
   a. One that accepts any active voice sentence using the nouns, articles, transitive and intransitive verbs, "and", and prepositions given in the lesson (over 17 million variations are possible)

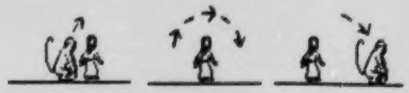
Figure 4: Summary of Lessons Developed

Other Suggested Lessons for Development

1. Sentences with adverbs
2. Sentences with passive voice
3. Sentences with adjectives
4. Lessons similar to the one developed above (Figure 4 #4.) but with fewer sentence types and vocabulary
5. Lessons that animate a sentence, display 2 or 3 choices and ask the student to pick the best sentence
6. Lessons that illustrate a sentence and ask the student to make a sentence for it
7. Lessons that give the student a sentence and ask the student to determine if it is syntactically correct

Figure 5: Extra Anticipated Activities

tions which were then used to produce worksheets and overhead transparencies. The student was expected to use the vocabulary that s/he had used in the instructional lessons although it was hoped that the student might experiment with other vocabulary as well. Figure 6 shows an example of this worksheet/transparency.



| | | | |
|---|---|---|---|
| the | over | under | from |
| girl | bird | frog | witch |
| runs | jumps | swims | turns |

Write 4 more sentences.

1.________________________
2.________________________
3.________________________
4.________________________

Figure 6: Sample Worksheet/Transparency

The transparencies were used as a group activity. First, a student would create the sentence for the given animation. Then, the students would alternately create similar sentences on the lines provided. The animations were done by the students themselves in the form of role playing. This activity aided especially in comprehending the use of prepositions which is particularly difficult for many deaf students to incorporate into their written English. Corresponding worksheets were then given as seat work or as homework.

Most of the students worked quite successfully within this structured framework. One student, however, showed some creativity by appropriately incorporating additional vocabulary into some of his sentences.

For the second need, two materials were developed: (a) a set of cards with rectangles on them similar to those used in the instructional part of the student's computerized lessons and (b) a set of worksheets with several sets of rectangles on them. In both instances, the students were asked to make sentences using the vocabulary from the lessons. The sentences could then be checked on the computer or by the teacher. Figure 7 shows an example of one of the cards.

Make a sentence.
Put words in the boxes.

☐ ☐ ☐ ☐ ☐

Did you make a sentence?
Try it on PLATO.

Figure 7: Example Card

A first set of cards with no vocabulary clues was given. After the student mastered these, s/he was given the second set which conatined one clue word. This was more difficult since the clue word limited the number of possible correct answers and, in most cases, required teacher assistance.

Another need also surfaced while the students were using these lessons. This was the need to check reading comprehension since, initially, many of the students could not read some of the words in the lesson; i.e., the prepositions and some of the verbs. This led to the development of still another reinforcement material. The vocabulary block was reconstructed by putting each word from it on a 3 x 5 card. The children were given the cards in a stack and asked to reconstruct the block from memory by placing the cards on a flat surface. They were then asked to arrange the appropriate cards into a sentence and to sign/mime the animation. A reversal of this procedure was also done in which the teacher provided the animation and the student had to construct the appropriate sentence. It was at this time that any actual instruction was done by the teacher if the need was indicated.

## PROCEDURES

The original strategy was a weekly group activity using the sentence lesson with the teacher and all five students. Game lessons were not a part of the exercise at this time. The students became impatient for their turns which resulted in frustration and arguing. After several weeks of this, it was obvious that a different strategy had to be found.

Use of a second terminal was added. The class could now be divided into two groups: one consisting of two students; the other, three students. The services of the PLATO aide were also employed to help supervise the students at one of the terminals while the teacher was stationed at the other. This supervision was passive in nature since one of the goals of the program was to promote independence. In addition, some game lessons were added to the students' indexes. The weekly time frame remained the same, but the quality of the time had been improved.

Both structure and flexibility were provided for the students in this second attempt. During the first 30 minutes the students used the sentence lesson to make sentences which they then recorded on paper. They took turns, making as many sentences within the 30 minute time frame as possible. At this point, it was noticed that the students began to offer assistance to one another in the form of explanations and demonstrations of the correct sentence type. Sometimes the help was accepted; at other times, the student would reject any assistance and experiment until PLATO accepted the sentence. These were the first signs of behavioral improvement. The students were beginning to work together successfully and, at the same time, to develop a desire to work independently.

The students could use the remaining 25 minutes to play one of the four games on the class index. Playing the games served a twofold purpose: (1) as a reward for hard work and good behavior and (2) as practice for learning how to accept defeat.

As the semester continued, the students were given the opportunity to work unsupervised during their PLATO time. They were able to sign on, complete their assignment and play a game within the 55 minute time frame without wasting time and energy on arguing.

In January, the strategy was changed again. The class had mastered the sentence lesson and was ready for more advanced work. This luckily coincided with the completion of the development of the new Animated Sentence lessons that were to be field tested at KDES. These lessons offered the student new vocabulary as well as more complex sentence formats. Since the students responded so well to the previous sentence lesson it was decided to increase the computer as shown in Figure 8 below:

|  | M | T | W | Th | F |
|---|---|---|---|---|---|
| Terminal 1 | 11:30<br>12:25 | 11:30<br>12:25 | 11:30<br>12:25 | 11:30<br>12:25 | 11:30<br>12:25 |
| Terminal 2 | 11:30<br>12:25 | 11:30<br>12:25 | 12:00<br>12:25 | 12:00<br>12:25 | 12:00<br>12:25 |

Figure 8: Revised Computer Schedule

The Monday and Tuesday sessions were still used for group activities; however, all 55 minutes were used for sentence work. This change was made because the students were now exhibiting a maturity level that could accept delayed reinforcement.

The Wednesday and Thursday sessions were divided into 15 minute segments in order to provide each student with an opportunity to work on the computer alone. This not only gave the students more independence, but also reduced the class size for more individualized instruction with the teacher. Friday sessions were reserved for the reinforcement time; i.e. the students were permitted to play the games if they had worked well during their Monday through Thursday sessions.

The student's indexes were also changed. They could now choose from eight different sentence types or the ninth choice, "Make Any Sentence", which would allow them to make any of the sentence types they had learned. After a few weeks, they had practiced all of the sentences and preferred to choose "Make Any Sentence" so that they were not limited to one particular sentence type. The procedure at the computer varied. The students were to either (1) do a particular sentence lesson as assigned by the teacher, (2) make any sentences they wanted, (3) use teacher made materials in conjunction with the computer, or (4) check classwork on the computer. All work was recorded by the student in a special notebook for computer work.

## RESULTS/DISCUSSION

### Academic

The academic growth in the area of Reading and English Language skills was a result of the entire educational program at KDES. However, the sentence lessons on PLATO were an important part of that program. Yet, it was difficult to identify any given area of growth as a result of a specific educational strategy or tool. The following are some areas of growth that have been seen in the students' Reading/Language performances which are directly related to the skills emphasized in the

PLATO sentence lessons:

(1) the ability to make more complex sentences within the structure of the PLATO lessons

September: The boy runs.
March The boy pushes the ball under the table.

(2) an increase in English vocabulary. All of the students could read and sign most of the words in the vocabulary box. Most of the students could use the verbs correctly but some of them still confused "push" and "pull". The same students also had trouble with "over" and "under" but with individualized instruction and extra materials, these problems were eliminated.

(3) an increase in reading comprehension within the structured lesson. As a periodic check of the students' comprehension of the sentences s/he created, the teacher would ask the student to predict the animation by signing or miming it before the computer animated it. Most of the time the prediction was correct.

(4) an increase in the use of English-like structures in the students' creative writing Obvious progress has been made within the structured sentence lessons on the computer. The students were able to read the vocabulary and to produce complex sentences. The next step was to see if any of the vocabulary appeared in their creative writing. Upon examination of the students' past and present writing, it was found that, in some cases, there was more usage of the preposition "to". Several writing samples have been selected to better illustrate this point.

Student 1
(then)
The man talk the woman.
All Indian go back the mountain.
(now)
James came to school at 8:30.
All of elementary go to rollerskate in Md.

Student 2
(then)
girl go school
dog run the boy
(now)
Tony go to store
Go to house

Occasionally some of the students with better English skills have begun to expand on the vocabulary within a sentence format. For example, one student used the following sentence in a story: "The boy play with the girl." Here, she used the preposition "with" correctly. Although this is difficult to determine, the student could be generalizing a sentence format learned in the Animated Sentence lessons. The important point is that there was improvement.

Non-Academic

Many of the immature behaviors previously described still existed but appeared less frequently in class. This was largely a result of the classroom behavioral system which required the child to earn the right to use the computer. The strong desire to work at the computer usually outweighed the need to act immaturely. The students' behavior at the computer itself was excellent. They were aware of the rules (for behavior) while at the computer and any infringement (i.e. fighting, refusing to share, doing an unassigned lesson) resulted in the loss of the privilege to use the computer for at least one session.

The students' abilities to work independently have increased tremendously. They no longer need the teacher at all for supervision. They are willing to experiment and do not become discouraged if they fail at the first few attempts. One boy, who normally was reluctant to write English showed a great deal of creativity on PLATO. He was highly motivated by the animations and experimented with the words until he was able to create a new animation. One such example is: "The boy runs over the car to the dog." This sentence includes two prepositional phrases which is a highly complex sentence for this child to make. He was also able to substitute different nouns to vary the animation. His enthusiasm encouraged other students to try to experiment as well.

## SUMMARY

In most settings where the use of computers as an educational tool is duscussed, the emphasis is on individualized instruction for one student and his/her resulting academic gains. The experience described in this paper indicates that other applications should be considered in order to fully utilize computerized educational programs. Among these applications for consideration are: (1) the use of one terminal by more than one student at a time; (2) the use of the computer as a reinforcer for desirable classroom behaviors; (3) the coordination of classroom activities so that they are compatible and mutually reinforcing of academic concepts taught; (4) the development and reinforcement of appropriate social behaviors such as taking turns and team work through teacher/adult guidance at the terminal. The combination of the usual academic considerations and those listed above should help to maximize benefits from computerized education.

## NOTES

1. PLATO is an interactive-graphics computer system that was developed by the Computer-based Education Research Laboratory at the University of Illinois and marketed by Control Data Corporation with headquarters in Minneapolis, Minnesota. It provides instructional material in over 100 curricular areas from early elementary school to the university level. More information can be obtained by writing: PLATO Services, 4201 North Lexington, ARH233, Arden Hills, MN, 55112.
2. For more information on these materials or others contact PLATO Services at the address listed above (#1).
3. Among the authors are: Sharon Dugdale, Dave Kibbey, Charles Weaver, and William Golden from the University of Illinois, Urbana, IL

61801; and Bonnie Seiler from the University of Deleware, Newark, DE 19711.

4. For more information on this project, write Phil Mackall, Kendall Demonstration Elementary School, Gallaudet College, Kendall Green, Washington, D.C. 20002
5. For more information on this project, write Phil Mackall at the address listed above(#4).
6. For a copy of the guidelines, write Phil Mackall at the address listed above (#4).
7. For more informatio on the Animated Sentence lessons, write Phil Mackall at the address listed above (#4).
8. Charrow, Veda R: A Psycholinguistic Analysis of "Deaf English"; Sign Language Studies, 7:141, 1975.
9. Stokoe, William C: The Use of Sign Language in teaching English: American Annals of the Deaf, August, 1975, p. 417.
10. For more information on "group notesfiles" contact PLATO Services at the address listed above (#1).
11. For more information on "mrouter" contact PLATO Services at the address listed above (#1).

# USE OF CAI IN A MAINSTREAMED HEARING IMPAIRED PROGRAM

Mary Garvey
St. Louis Park Senior High School
6425 West 33rd Street
St. Louis Park, MN
HENNEPIN TECHNICAL CENTERS

In September, 1980, a CAI program was initiated as a supplement to special class and resource room instruction in the St. Louis Park Hearing Impaired Program. Systems presently include four Apple II microcomputers and two Teletype 43 s. Software distributed by MECC and TIES, as well as staff developed programs are in daily use. Tutorial and drill and practice programs supplement instruction in language, vocabulary, reading and math development programs within special classes. In the resource rooms, CAI provides drill and practice on vocabulary and concepts from such mainstreamed classes as math, science, health, home economics, English, social studies, and career exploration. Computer time has quickly become a sought after reinforcer in behavior management programs.

As part of the Hennepin Technical Centers staff, special education professionals serve preschool through post-secondary students with low incidence handicaps in 13 west metropolitan school districts and Minneapolis. Approximately 55 severe to profoundly hearing impaired secondary age students attend centrally located mainstreamed day school programs at St. Louis Park Junior and Senior High Schools. This population includes students with academic functioning ranging from minimal to gifted, some of whom have additional learning problems such as severe vision impairment, severe emotional disorders, low English proficiency, and one user of the Bliss Symbolic System for expressive/receptive language. Services are based on the student's IEP (Individual Educational Plan). These may include self-contained classes in English, reading, social studies, speech and language, as well as academic support for mainstreamed classes in the form of parallel teaching, team teaching, resource support, or interpreting. Specialized auditory or visual teaching strategies, communication modes and/or equipment are provided.

Deaf students require special instructional techniques because of the cumulative effect of hearing impairment in the process of education. The absence of auditory input significantly affects development of language and speech. Language is basic to communication, and communication is basic to learning and socialization. The most dramatic illustration of the impact of language deprivation is in the areas of reading and written language, the cornerstones of education. At the secondary level, when learning materials and concepts are presented in increasingly more sophisticated language, the average deaf student has attained a reading level of between third and fifth grade and a vocabulary approximating that of a third grader. Without intense formal work he will have difficulty both in comprehending and generating relative clauses, idioms, passives and compound and complex sentences. He will struggle with generalizing the formal language rules he has mastered and will find critical reading skills elusive.

In the mainstream, therefore, a great deal of time is spent on mastering the language of instruction as well as the concepts presented. Drill and practice on vocabulary, spelling and language structures are part of both self-contained classes and academic resource support. Over the span of education the hearing impaired student tends to become more rather than less dependent on special intervention because of the cumulative language deficit. At the secondary level, when he is striving for the same independence as his hearing peers, his motivation suffers when the learning task nearly always includes an adult who seems to be forever hovering about with the same drill and practice routines. In spite of good intentions, we are less than successful as special educators, I believe, if we do not address both the intervention and independence issues.

My interest in Computer Assisted Instruction as a tool for both intervention and independence was kindled at a February, 1979 ADCIS convention. There were at that time a few timeshare and dedicated systems, such as PLATO, CCC and TICCIT, offering a limited number of lessons designed for the hearing impaired in reading, language and math. The prohibitive cost of these systems made them impractical unless a program 1) had a large number of students who would benefit and 2) had federal or private funds to support such a venture. With neither at hand, I filed the interest and information away until fall, when we began preliminary planning for more than double the population of hearing impaired students on the secondary level.

My proposal that we apply for a federal grant for CAI happily coincided with burgeoning microcomputer technology and a high degree of administrative interest in the use of computers within the district. In the process of planning it was decided that the district itself would fund the hardware and a half-time facilitator position. Minnesota already boasts a sophisticated system for development and dissemination of educational computing materials in MECC and TIES. It was felt that tying in with these existing facilities and using the first year as a time for exploring the market for suitable software would be the most cost effective means to building a CAI program.

My initial request for four 32K Apple II microcomputers and two Teletype-43 on-line systems and supplies was built into the 1980-81 budget. Hardware was purchased as much as possible on state contracts, with delivery set for fall, 1980. Membership in MECC and TIES provided access to software as well as training in use, course development and programming of both types of systems. Members are also allotted timesharing on the MECC CDC Cyber system and TIES Hewlett-Packard system.

We do not purport to be anything but neophytes in the area of computer assisted instruction. Our goals this year have focused on orientation, integration and exploration. Teachers who are not oriented to what CAI is will not use it. They must be convinced that it is not a replacement for a skilled teacher, but a very powerful support tool which can enhance individualized instruction. MECC and TIES offer workshops and classes to member districts which give a well organized orientation to hardware and software, and provide development training. It is part of the program facilitator's job to be current on available literature and learning opportunities and promote these among the staff.

We have not yet done any formal student orientation. Basic instruction in use and "non-abuse" of the systems was readily accomplished in small group or individual sessions. We do plan to incorporate a short course on CAI orientation into our prep class for new students in the fall. Students wanting to take programming or computer literacy courses are encouraged to sign up for the regular mainstreamed classes.

We have found that making hardware and software easily accessible to both self-contained and resource sessions is key to integration of CAI into the program. Our Apples are scheduled into self-contained classes and arrive via the facilitator or student aides. Having the computer in the classroom allows the classroom teacher to more closely monitor the students' progress and eliminates "wasted time" when students move from classroom to CAI area.

The Apples "live" in the resource rooms when they are not traveling to self-contained classes so that drill and practice sessions on vocabulary and concepts may be easily incorporated into a resource session. The on-line systems are housed in a CAI area and are scheduled and proctored by the facilitator.

A card system for cataloging available software by name or subject matter has made it easier for teachers to incorporate CAI into their lesson planning. Knowledge of the particular students' learning styles gained from participation in the class or resource session helps the facilitator pinpoint the most efficient application of the computer. Lesson development then becomes a team effort by the classroom teacher and the facilitator.

As you might guess, if you have any experience working with the hearing impaired, exploration of available software has been disappointing. The areas in which our greatest needs lie: reading, vocabulary and language, are largely undeveloped. The few commercial programs in reading and language arts are either unsuitable for the hearing impaired or are expensive and of limited use.

We depend heavily on the MECC and TIES authoring programs to develop drill and practice lessons for reading, English, social studies, science, health, home economics, math and career exploration. Lessons in language, vocabulary, spelling and concepts in these areas have added motivation and more individualized instruction to self-contained and resource sessions alike. Our newly purchased Apple Pilot will be used to develop tutorials in reading skills and language.

Many existing MECC and TIES programs for the Apple are in use with some adaptations. Rewriting or preteaching instructions and adding visual aids can make that "almost perfect" program suitable for hearing impaired students who have trouble with reading or instructional language. Computer games have quickly become sought after reinforcers for behavior management programs.

We hope in the near future to upgrade Apple II equipment to 48k and add better graphics capabilities. A course development training program tied to the hearing impaired reading, English and social studies curriculums should produce some significant software and further integrate CAI into the program. During the initial period, knowledge of efficient uses for CAI in the classroom/resource center, and enthusiasm over its effectiveness, has grown dramatically among our staff. In addition, a preliminary informal survey of program directors indicates interest in a CAI regional consortium to develop and disseminate courseware specifically for the hearing impaired.

In summary, CAI has proven to be a valuable supplement to special class and resource room instruction in our mainstreamed hearing impaired program. Software can be adapted or created relatively easily using authoring programs. Of particular importance is the accessibility of the systems to teachers and students, and optimal integration of CAI into the curriculum.

# LARGE TIME-SHARING MICROCOMPUTERS
# IN
# SECONDARY EDUCATION

James Gockel
St. Peter, Minnesota
Independent School District 508

This paper describes a proposal Independent School District 508 (St. Peter, MN) submitted to acquire a large microcomputer to provide part of its instructional computing capability and to perform some selected administrative applications. Since this proposal was written as a Title IV-C project, the project's objectives are given along with the accomplished objectives after the system was on-line six months.

The primary purpose of my proposal was to acquire a large microcomputer suitable for performing the following instructional and administrative objectives:

1. Additional on-line instructional terminals
2. Word processing for both the instructional program and the administrative program
3. Building security through key control
4. Statistical programs for guidance people
5. Audio-visual inventory and control
6. Continue to use the MTS and MIS systems
7. Provide an inventory of fixed assets
8. Provide a student record system
9. Provide a student data base for school census reports

The computer needs of ISD 508 were established by current demand for computer services at both the instructional and the administrative levels of this district. Needs were established and supported by the district Computer Advisory Board which consisted of school board members, computer science instructors from local colleges and local business people. These needs as established by this advisory board are as follows:

A. Instructional Computer Service:
    1. To provide GIS, Compute and other programs available on the MTS computer system
    2. Expand the number of instructional programs throughout the JR-Sr high curriculm
    3. Add additional terminals for mathematics and computer science
    4. Show application of computer literacy through demonstration of a time-sharing system
    5. Introduce word processing in modern office practice and journalism classes

B. Administrative Computer Service:
    1. Finance and Payroll
    2. Student grade reporting

3. Census

4. Word processing

5. Daily attendance

6. Maintain building security through key control

7. Inventory and control of Audio-Visual equipment

8. Statistical analysis

9. Student enrollment projection

10. A data base of all people zero through 20 years of age

11. Inventory of fixed assets

The following chart shows services that were available to meet our needs. Vertical axis matches subdivisions II-A and II-B. Horizontal axis shows 4 different options available to St. Peter. These options show what we feel could have been done at St. Peter with one or a combination of options. The option selected is the one we feel would best provide effective and efficient data processing at St. Peter Schools. If the option of not available (NA) is given, the task we wish to perform is not available on any system available to us. Options are left to right:

1. Minnesota Time-Sharing plus Minnesota Information Systems only

2. Minnesota Time-Sharing, Minnesota Information System, and Mankato State University plus small micro computers

3. Minnesota Time-Sharing, Minnesota Information System and Mankato State University plus both large and small computers. The small micro's (Apple II) would be used in the Elementary building

4. NA indicates system not available as an option

OPTIONS AVAILABLE TO ST. PETER

| | INSTRUCTIONAL NEEDS | | | |
|---|---|---|---|---|
| | MTS<br>MIS | MTS/MIS MSC<br>Sm. Micro | MTS/MIS MSC<br>Lg. Micro | MTS/MIS MSC<br>Lg/Sm Micro |
| II A-1 | Yes<br>NA | Yes<br>NA | Yes<br>NA | Yes<br>NA |
| II A-2 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |
| II A-3 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |
| II A-4 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |
| II A-5 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |

| | ADMINISTRATIVE NEEDS | | | |
|---|---|---|---|---|
| | MTS<br>MIS | MTS/MIS MSC<br>Sm. Micro | MTS/MIS MSC<br>Lg. Micro | MTS/MIS MSC<br>Lg/Sm Micro |
| II B-1 | NA<br>Yes | MIS<br>NA | MTS<br>NA | MTS<br>NA |
| II B-2 | NA<br>NA | MSC<br>NA | MSC<br>NA | MSC<br>NA |
| II B-3 | NA<br>NA | MSC<br>NA | MSC<br>NA | MSC<br>NA |
| II B-4 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |
| II B-5 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |
| II B-6 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |
| II B-7 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |
| II B-8 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |
| II B-9 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |
| II B-10 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |
| II B-11 | NA<br>NA | NA<br>NA | NA<br>NA | NA<br>NA |

Our alternatives for meeting the needs of ISD 508 as listed in section III and are given in the following chart. Although a small micro will perform some of the tasks listed, it is our belief that the large micro computer can perform these tasks more efficiently and could be more cost effective. Matrix code is the same as section III services now available to meet needs.

|  |  |  |  |  |
|---|---|---|---|---|
|  | INSTRUCTIONAL ALTERNATIVE |  |  |  |
|  | MTS<br>MIS | MTS/MIS<br>MSC<br>Sm.<br>Micro | MTS/MIS<br>MSC<br>Lg.<br>Micro | MTS/MIS<br>MSC<br>Lg/Sm<br>Micro |
| II A-1 | Yes<br>NA | Yes<br>NA | Yes<br>NA | Yes<br>NA |
| II A-2 | Yes<br>NA | Yes<br>Yes | Yes<br>Yes | NA<br>Yes |
| II A-3 | Yes<br>NA | Yes<br>Yes | Yes<br>Yes | NA<br>Yes |
| II A-4 | NA<br>NA | NA<br>NA | NA<br>Yes | NA<br>Yes |
| II A-5 | NA<br>NA | NA<br>NA | NA<br>Yes | NA<br>Yes |
|  | ADMINISTRATION ALTERNATIVES |  |  |  |
|  | MTS<br>MIS | MSC/MTS<br>MIS<br>Sm.<br>Micro | MTS/MIS<br>MSC<br>Lg.<br>Micro | MTS/MIS<br>MSC<br>Lg/Sm<br>Micro |
| II B-1 | NA<br>Yes | Yes<br>NA | Yes<br>NA | Yes<br>NA |
| II B-2 | NA<br>NA | Yes<br>NA | 1980-81<br>Yes<br>NA | 1980-81<br>Yes<br>NA |
| II B-3 | NA<br>NA | Yes<br>NA | 1980-81<br>Yes<br>NA | 1980-81<br>Yes<br>NA |
| II B-4 | NA<br>NA | NA<br>NA | NA<br>Yes | NA<br>Yes |
| II B-5 | NA<br>NA | NA<br>NA | 1980-81<br>NA<br>Yes | 1980-81<br>NA<br>Yes |
| II B-6 | NA<br>NA | NA<br>NA | NA<br>Yes | NA<br>Yes |
| II B-7 | NA<br>NA | MSC<br>NA | MSC<br>Yes | MSC<br>Yes |
| II B-8 | NA<br>NA | MSC<br>NA | MSC<br>Yes | MSC<br>Yes |
| II B-9 | NA<br>NA | MSC<br>NA | MSC<br>Yes | MSC<br>Yes |
| II B-10 | NA<br>NA | NA<br>NA | NA<br>Yes | NA<br>Yes |
| II B-11 | NA<br>NA | NA<br>NA | NA<br>Yes | NA<br>Yes |

Because a large micro system provided us with such things as large off-line storage, expandability, word processing, sort and merge utilities, screen oriented editor and many other advantages, we elected to go with option IV. However, it still was our intent to provide teachers K through 12 with small transportable micro computers. The following list supplies background for our proposal of a large microcomputer:

1. Important characteristics of the large micro in this proposal are multi-tasking, multiuser, large data storage, expandability, compatibility, data word processing, simple data base management system, many utilities such as sort and editor at reasonable cost.

2. The primary function of this proposal will be to provide additional instructional accesses while at the same time greatly expanding our administrative date processing capabilities that will not be provided for by MIS Region V. The system will provide 6 I/O ports of which one will be dedicated to administrative processing.

3. We are continuing to use MTS for information retrieval and as a software supplier for instructional programs.

4. All support and software conversion will be the responsibility of the district.

Our proposal was approved by the MECC board as a Title IV-C project but without funding on April 17, 1980. In July, bids were let and in August Microprogramming of Burnsville, MN., delivered a large microcomputer. The system includes an Alpha Micro processor with 128k of memory and a CDC Hawk disk drive which contains 5 megabytes of fixed storage and 5 megabytes of removable storage. The system is capable of handling 6 ports which include:

1. The console
2. An office terminal
3. A 150 CPS printer
4. One student terminal
5. One student terminal
6. A card reader or MECC Time-sharing terminal

Under the Title IV-C project, two objectives were to be achieved with four major activities under each objective. These objectives and their activities are as follows:

I. Administrative Objective

   1. Install and use the computer vendor's word processing software to produce at least one administrative document
   2. Load, store and statistically analyze the standardized test scores for grades 9-12 for the past 10 years
   3. Convert the grade reporting system from the Mankato State computer
   4. Convert the census system from the Mankato State computer
   5. Develop and implement a data base system containing census information for 0-20 year age group

II. Instructional Objective

1. Install three instructional terminals, a card reader and a printer to be used by computer science classes for teaching multi-programming languages, data base management and job control language

2. Relocated the MTS terminal and train staff to increase the use of GIS

3. Reallocate usage of MTS to increase use of computer on non-computer science curriculum areas

4. Convert at least one medium size MTS library program to assess the level of effort and storage required

After six months, all activities under the administrative objective and three of the four activities under the instructional objective were accomplished. We are very happy with the system and feel that all activities of both objectives as described in the Title IV-C project will be completed by years end. As more local funding becomes available, we hope to increase both the number of instructional as well as administrative terminals. Interface software is being sought to tie the Apple II computers and the MIS B6800 computer to the Alpha Micro system. Because the majority of the administrative data processing is now being done on the Alphy Micro, administrative costs have decreased while the instructional programs have increased due to additional time-sharing terminals.

# IMPLEMENTATION AND EVALUATION OF A COMPUTER-ASSISTED COURSE IN MUSICAL AURAL SKILLS

Dorothy Gross and Wendy Griffin
Music School, 105 Scott Hall
University of Minnesota
Minneapolis, Minnesota 55455

Studying musical aural skills often requires a partner for effective learning. Since it can be difficult for students to arrange convenient practice sessions, computers offer an alternative to a human partner. We selected the Apple II microcomputer because it has a tone generator, some existing music software, and, with the addition of an ALF board, could produce up to three simultaneous tones. Several programs, including one original one, were used. Altogether, the courseware covered intervals, scales, and melodies. Evaluation of this project, including attitude questionnaires and achievement tests, is largely positive and suggests that the use of computers should be expanded.

Aural skills form part of the core music-major curriculum at the college level, and for many students present great problems. In part, the methods of practicing outside of class have made effective practice difficult. Specifically, the most flexible and recommended technique has been to drill with another student. Unfortunately, students often have problems arranging convenient times and places for such work, and often complain of the lack of opportunity to prepare outside of class. As an alternative, tapes have been used at many schools, but recorded material has limitations. Most importantly, it lacks the flexibility to adjust to individual needs, a prime requirement in a subject fraught with individual differences. Second, students get very little guidance or analysis of their errors. Clearly, another method of ear-training exercise is desirable, combining the availability of tapes with the individualization of a partner or tutor.

Computers seemed to offer an attractive alternative to the traditional drilling methods described above. Properly programmed, they could provide personalized instruction at the students' own pace, and even include competency-based sequences (Foltz and Gross). Through advance scheduling, a computer would be available at convenient times and places.

## IMPLEMENTATION

After examining many possibilities, we selected the Apple II microcomputer since it had the potential to meet our instructional and musical requirements. In particular, effective ear-training requires a computer that can generate and play music. The Apple II has a built in generator for single tones, and with the addition of a music board, can play several tones at once. The ALF board, capable of playing up to three simultaneous notes, was chosen because existing music courseware was available for that board.

The next step, after acquiring the hardware, was to preview, and ultimately select, software. We decided, rather than making our computer laboratory a library of available courseware, to select one drill for each aspect of the course. In doing so, we were able to compile a single set of lessons, coordinating with our curriculum.

The final lessons available to students were interval drills, melodic dictation, scale drills, and pitch patterns. The interval drills, written by Charles Boody, included melodic and harmonic intervals. With these drills, students could control many parameters, and also received help in understanding and correcting their errors. Another of Boody's drills permitted a competency-based approach to the same subject.

The melodic dictation programs, written by Dorothy Gross (Gross), actually included melodic, rhythmic, and pitch dictation, based on stored melodies. Students selected the type of drill and level of difficulty (see Figure 1). Then the program scored the dictation and recommended the next level, based on the students' pitch and rhythm scores.

Several programs by Linda Borry (Borry) were used, particularly for scales and chords. In addition, her "Wrong Note" drill gave the class practice in pitch error detection, as students saw a line of music displayed on the video monitor, heard the music played with one error, and had to identify the mistake. A similar program, "Missing Note," presented students with a line of music with one note missing. Students listened to the example and added the missing note.

The class used as a pilot group was a five-week review class in freshman ear-training at the University of Minnesota. Students met in class for five hours a week, learning aural recognition of intervals, scales, chords, melodies, and progressions. Of these subjects, all but the last were supplemented by computer drills. Although seventeen students took the course, one missed the pre-test, so the remaining sixteen have been used for evaluation purposes.

Before beginning actual student use, we prepared instructions to the students in computer operation. These instructions were followed up by group demonstrations in order to familiarize students with the equipment. Finally, we drew up a schedule, assigning each student two tenty-minute units a week at the Apple II. Some extra time also was available on a first come, first served basis.

EVALUATION

Identical pre- and post-tests were administered measuring students' knowledge of the material covered in the course. The scores then were compared statistically to the number of twenty minute units that each student spent at the computer. Attitude surveys were conducted at the beginning and end of the course. Additionally, we informally observed students' behavior and attitudes as they practiced at the computer.

Statistical analysis of the results of the experiment was accomplished through the use of MULTREG, a regression program, at the University of Minnesota (MULTREG). An evaluation of whether computer time had an effect on post-test scores, after adjusting for pre-test scores, was analyzed by regression of the post-test scores on the pre-test scores and units of computer time. The p-value was approximately 0.06, just shy of statistical significance. Since statistical significance is more difficult to attain with a small sample size, such as ours, these results suggest a positive relationship between time spent at the computer and progress in the course.

Student responses to the post-class survey about the use of the computer were largely favorable. Advantages to using the computer reported by students included the ability to drill and test on their own, the patience of the computer in contrast to a partner, immediate playback of errors, the ability to individualize instruction by intense practice in

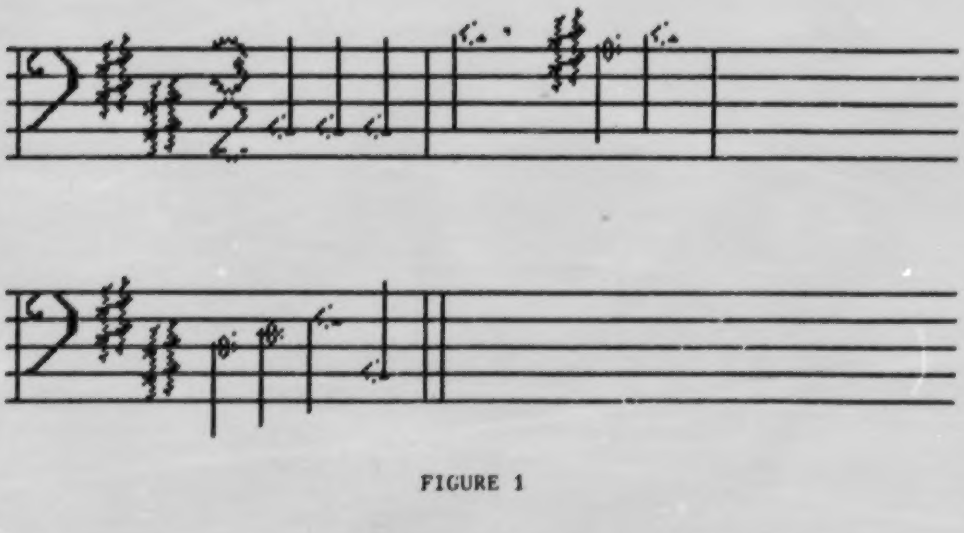


FIGURE 1

problem areas, quick response from the computer, and error charts that indicated areas of confusion. One student particularly liked the privacy of computer practice, because only the computer knew his errors. Three students have continued to use the computer for drilling, even though their coursework no longer requires it. Especially noteworthy is that prior to experience with the computer in this class, 53% of the students reported liking computer in general. After experience with the computer in this class, 80% of the students reported liking computers in general. Responses to the post-class survey asking for disadvantages to using the computer included the artificial-sounding tone quality, difficulty using the computer for non-typists, and the problem of computer malfunctions due to overheating.

Despite these complaints, the computer was almost continuously in use. In fact, the most common complaints were that more computer time, more programs, and more computers were needed.

In summary, it appears that using our computer assisted instruction sequence in an ear-training class had a positivie effect on improving students' aural skills, and should be expanded beyond the scope of our study. Further investigation might include a measurement of the effect of other variables, such as class attendance and other methods of drill outside of class. Additional data might be accumulated regarding the time spent by each student on each program. In any event, we will continue to use and encourage CAI in our aural skills instruction.

----------

## REFERENCES

Boody, C. Interval drills. Programs on the Apple II Microcomputer, 1979.

Borry, L. MECC Music Disk. Programs on the Apple II Microcomputer, Apple Corporation, 1980.

Foltz, R. and Gross, D. Integration of CAI into a music program. Journal of Computer-Based Instruction, 1980, 6(3), 72-75.

Gross, D. Simple Simon. Programs on the Apple II Microcomputer, University of Minnesota, 1980.

MULTREG. Program of the Applied Statistics Department, University of Minnesota.

## ACKNOWLEDGEMENTS

Funding for this project was provided by the University of Minnesota. We acknowledge the particular help of Dennis Jennings of the Statistical Clinic at the University of Minnesota.

# AN EMPLOYMENT APPLICATIONS INFORMATION SYSTEM

*Charles G. Hadduck*
*Systems Manager*
*Portland Public Schools*
*501 N. Dixon*
*Portland, OR 97227*

*The Employment Applications Information System recently implemented at Portland Public Schools is described and demonstrated. This on-line system allows the Personnel Department to input pertinent data from all employment applications into the computer. Inquiries, based on various selection criteria, can then be made via display terminals to obtain information regarding job applicants. The system helps assure that all qualified applicants are considered for each job opening. A variety of printed reports are also provided to assure affirmative action compliance, for statistical analysis purposes and other uses.*

Over the past several years, financial constraints and declining enrollment in many urban school districts have created unique problems in the hiring of teachers and non-certificated personnel. Reduced funding has lead to the requirement for hiring teachers who are qualified in several subjects as well as possibly having one or more coaching endorsements. The requirements of Title IX and Affirmative Action lead to even further combinations that must be considered. While declining enrollment has meant that fewer new teachers are needed each year, it also means that many more teachers are applying, and thus many more applications must be evaluated.

Manually reviewing a large file of applications looking for an English teacher who is also qualified to teach Biology and can coach women's volleyball and golf is a slow and error-prone process. Multiply this process by several hundred which might well be the number of new teachers required in a district during any one year, and the task becomes staggering. Non-certificated applicants have fewer variables to be examined yet still require extensive analysis to satisfy all legal considerations. Responses to questions such as: How many minority applications did we receive? How many were offered jobs? How many accepted? Is this an improvement over last year? Requires an equally time-consuming effort, and usually when the response is urgently needed.

The Personnel Department asked Data Processing if it could help with these problems and a systems design was jointly developed with programming done by Data Processing over a period of several months.

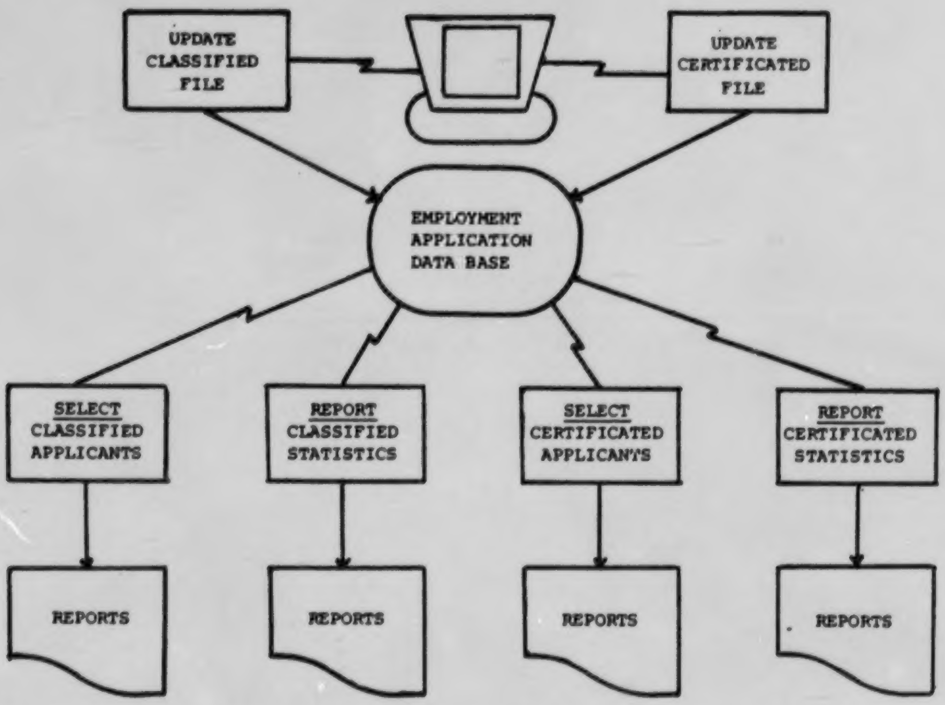
## INTRODUCTION

The Employment Applications Information System developed by Portland Public Schools is an online system that allows the Personnel Department to input pertinent data from all employment applications into the computer. Inquiries, based on various selection criteria, can then be made against the data base via display terminals to obtain information regarding job applicants. The system helps assure that all qualified applicants are considered for each job opening. A variety of printed reports are available to assure affirmative action compliance, for statistical analysis purposes and other uses.

This system is installed on the host computer of Portland Public Schools, a Honeywell DPS440 with 512k words of main memory and 2.5 billion bytes of on-line disk storage using TPE, and COBOL 74 under GCOS 3. A variety of display terminals are used throughout the district with the Harris (a Honeywell VIP compatable) terminal most widely used. The Employment Applications Information System is designed to use the features of the Harris terminal.

## SYSTEM DESIGN

The Employment Applications Information System is a series of six programs which update the application data base, select report information in accordance with the allowable variables, and print the selected information. The data base is an indexed random file consisting of two record types; a 500 character Certificated Applicant record, and a 350 character Classified Applicant record with the index key being social security number. The conceptual view of the system is as follows:


UPDATE CLASSIFIED FILE
UPDATE CERTIFICATED FILE
EMPLOYMENT APPLICATION DATA BASE
SELECT CLASSIFIED APPLICANTS
REPORT CLASSIFIED STATISTICS
SELECT CERTIFICATED APPLICANTS
REPORT CERTIFICATED STATISTICS
REPORTS
REPORTS
REPORTS
REPORTS

## INPUT

All data is entered into the computer by the Personnel Department. When the specified Keyword is entered, a form is displayed on the terminal screen. A new application may be entered by completing the screen form with information from the application form and then transmitting the variable data from the completed screen form to the computer. An application that is ready on the file may be updated by entering the social security number. The screen form is then displayed with the information that is on file for the specified applicant. This information may be changed and then transmitted back to the data base. The transmitted data will replace that which was previously on file. If an applicant has been hired or is no longer interested in being a candidate for employment, an inactive code may be entered while in the update mode. The application will remain on file, for statistics purposes, for a period of time specified by the Personnel Department and coded in the update program.

## OUTPUT

Select List of Applicants

When the specified keyword is entered into the computer, the selection screen form is displayed on the terminal screen. The user completes the screen form with any combination of selection criteria for which they wish to find applicants.

When the selected screen form has been completed and transmitted, a report is printed that includes the social security number, name and phone number for each applicant that meets the selection criteria. The report is concluded with a synopsis of the selection criteria.

## STATISTICAL REPORTS

Statistical reports may be generated by entering a program report name and then answering several questions pertaining to job categories. Specifific or general job categories may be specified. Based on the specified category, a listing of applicants by ethnic and sex code is printed. This report displays number of jobs, whether offered, accepted, rejected, and what percentage each is of the total number of jobs.

Selection Lists and Statistical Reports are then directed to a printer that is located in the Personnel Department.

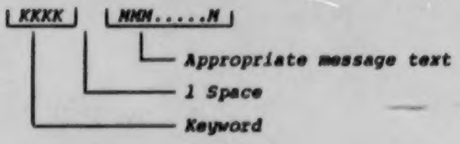
## SYSTEM EXAMPLE

1. Log on to TPE (Transaction Processing Executive) using the Harris terminal.

   - Type in the log-on message $*ID,24,HI-TRX and depress TRANSMIT. The computer will respond with the TRAX mm/dd/yy hh.mmm when the connection is made.

   - When the computer returns with a question mark (?), enter your initials (e.g., CGH) and depress TRANSMIT. The systems acceptance response is ****. Your terminal is now known to the system by your initials.

   - Enter your transaction message in the format.



   (a menu of keywords and message texts is available to system users (e.g., PAP4 UPDATE))

   When a satisfactory message has been input, the system will acknowledge with a sequence number in the form of #nnnnn which is used to trace transactions if there is a problem.

2. UPDATE an employee application record

   - Log on to TPE

   - Enter the keyword and transaction message (e.g., PAP4 UPDATE).

   - A message asking for the social security number will be displayed on the terminal.

   - Enter the keyword (e.g., PAP4) followed by the SS#.

   - The screen format with the requested employee will appear and the form is now ready for use. The TAB key is used to move the cursor from one field to another. Data can be entered or replaced as appropriate. (See Figure 1).

   - The cycle may be repeated as many times as desired. When finished, type DONE and press the TRANSMIT key.

3. Select applicants which meet criteria variables.

   - Log on to TPE

   - Enter the keyword and transaction message (e.g., PAP5 START). A tutorial procedure will initialize the screen format for proper operation.

   - Enter the keyword and transaction message (e.g., PAP5 SELECT) and select from the option menu all the selection variables against which you wish to interrogate the file. Selection variables include the following:

     *Report type, Category of Job, Sex, Ethnic Group, Time Basis, Handicap, Grade Level, Skill Code, Activity Code.*

   - When entries are visually verified, depress the TRANSMIT key.

   - If additional selects are desired, the above steps are repeated. Then enter the keyword and transaction message (e.g., PAP5 DONE) and press the TRANSMIT key.

   Reports as requested are generated into a print file which can be displayed on the terminal or directed to the attached hardcopy printer. Figure 2 is a sample report showing all applicants who met the desired criteria. In this example, the selection variables were Report type 1 (all applicants on file), Category 5 (all categories), Ethnic 1 (American Indian).

4. Report of statistical relationships for all employment applications.

   - Log to Honeywell Time Share

   - Enter /PAP600H

   - The computer will ask a series of questions relating to the categories you wish to include in your statistical report. An example of the screen formatted tutorial and the resultant report is shown in Figures 3 and 4.

In the examples discussed above, the Classified category of applicants was used. Each application has been entered into the file with appropriate coding as required by the system. Thus, the viability of the system rests on the fact of complete and up-to-date information being entered. As the file grows and the data monitored for accuracy, the information becomes a useful tool for quickly obtaining a list of applicants with specialized skills or applicants with a desired combination of skills.

```
KEYWORD AND OPTION STRING -
S.S.NBR 536-58-1509         APPLICATION D ATE (MM/DD/YY) 08/15/80
NAME (LAST,FIRST,MIDDLE) ASHAMALLA ALPHONSE ELIAS
ADDRESS: STREET 5617 G SE 88                  CITY  PORTLAND
........STATE OR  ZIP 97266
PHONE (503)    -7711          MESSAGE PHONE (503)    -7711
DATE OF BIRTH (MM/DD/YY) 06/24/41
.
TIME BASIS (F,P,S,10 OR 12) F  CITY LOCALITY (N,NE,NW,SE,SW) SE
RACE (1-5) 4 SEX (M,F) M HANDICAP (YES OR NO) NO  IF YES            CONV N
POSITION CODE 1.       2.       3.       4.       5.
PLACEMENT: ED       EXP       TRAIN       APP       ATT
AIDES: (GRADE LEVEL,SKILL) 1. ,      2. ,      3. ,      4. ,      5. ,
.
*JOB OFFER OR ACCEPTED FORMAT: XYZZZ/MM/YY. REJECTED FORMAT: RSN/XYZZZ/MM/YY
JOBS: OFFERED 1.    /  /   2.     /  /   3.     /  /
......ACCEPTED (XYZZZ/MM/YY)       /  /
......REJECTED (RSN/XYZZZ/MM/YY) 1.  /     /  /   2. /     /  /
---------- 3.  /     /  /
ACTIVITY CODE (ENTER X=INACTIVATE,D=DELETE OR LEAVE BLANK)
```

(FIGURE 1)

```
                                        REPORT DATE 03/02/81
                   APPLICATIONS


 SS NO.               NAME                          CONTACT PHONE NBR.

001-00-0001        WHITE FOOT, CHIEF ALBERT             503-222-2222
001-80-1275        BEEGLE, ROBERT, JR                   503-222-6960
001-50-1961        PULLEN, TERRY ANN                    503-222-2011
001-58-9587        CROSS, BONNIE LEE                    503-222-0606
001-62-7508        CRAWFORD, LONNA JEAN                 503-222-2481
001-66-3633        BATTEN, MICHELLE GEORGEANN           503-222-3644
001-88-5012        ASHBY, RODNEY WILLIAM                503-222-5197
001-88-6751        CRUS, LARRY L                        503-222-8380

** OPTIONS INCLUDED IN ABOVE REPORT ARE :
     SEX        ETHNIC 1      TIME BASIS         HANDICAP
     AIDES : GRADE LEVEL    SKILL               ACTIVITY
```

(FIGURE 2)

```
                    CLASSIFIED APPLICATIONS                  RUN 03/02/81
                       --- ALL JOBS ---                      FOR DATES 01/80-02/81
                       Job applications

               ---------MALE---------   --------FEMALE--------   --------TOTAL---------
                 MTD         YTD          MTD         YTD          MTD          YTD
American
   Indian        0   0%      6    1%      0    0%     4    0%      0    0%     10    2%
Asian            0   0%     10    2%      0    0%     5    1%      0    0%     15    3%
Black            0   0%     48   10%      0    0%    50   10%      0    0%     98   21%
Hispanic         0   0%    116   25%      0    0%   147   32%      0    0%    263   57%
White            0   0%      8    1%      0    0%     6    1%      0    0%     14    3%
Unknown          0   0%     14    3%      0    0%    42    9%      0    0%     56   12%

     Total       0   0%    202   44%      0    0%   254   55%      0           456
```

(FIGURE 4)

```
*** CLASSIFIED APPLICATIONS REPORTS ***                03/02/81    9:11

*************************** CLASSIFIED APPLICATIONS *****************************

Applications:   458     FOR DATES: 01/79-01/81
Jobs offered:  NONE
Jobs accepted: NONE
Jobs rejected:    1     FOR DATES: 06/80-06/80


What time period do you want the report to cover?    (MM/YY - MM/YY)

Enter the period beginning date:
?
*************************** CLASSIFIED APPLICATIONS *****************************

Applications:   458     FOR DATES: 01/79-01/81
Jobs offered:  NONE
Jobs accepted: NONE
Jobs rejected:    1     FOR DATES: 06/80-06/80


What time period do you want the report to cover?    (01/80 - MM/YY)

Enter the period ending date:
?
0281
Do you want to list:

          1.. Jobs applied for

          2.. Jobs offered

          3.. Jobs accepted

          4.. Jobs rejected


Select one or more of the options.
    (If you select more than one, separate the choices with commas)

Enter:
?
1______

You can get totals for all jobs..  or, you can count only
jobs in a general category (X), a specific category (XY), or
for one specific job (XYZZZ).

Choose one:
             1... ALL
             2... X
             3... XY
             4... XYZZZ

Enter:
?
1

The report is finished.

To see the report at the terminal, type "LIST *L".

*End of Job
```

(FIGURE 3)

# A MICROCOMPUTER MANAGED
# INSTRUCTIONAL SYSTEM

Gregory C. Hamilton
Office of Minority Student Education
College of Engineering
Michigan State University

This paper describes a research project investigating the uses of a microcomputer for 1) the construction and presentation of instructional materials, 2) recording and analyzing student responses, 3) the diagnostic prescription of subsequent instructional units, and 4) the management and sequencing of materials on an individual student basis. A system has been designed to integrate all of the above functions, and following is a discussion of its capabilities and limitations. Several utility programs used to facilitate the development of instructional materials are described. In addition, a program developed for the actual preparation and storage of instructional materials, including an author language, is presented.

## Introduction:

This paper describes a three phase educational research project being conducted by the Office of Minority Student Education in the College of Engineering at Michigan State University. The project, entitled a Microcomputerized Instructional, Testing and Evaluation (MITE) system, is designed to be an instructional management program. Although the intended use of the system is to provide individualized math and science instruction to freshmen and sophomore engineering students, the system itself is designed to be subject independent. The three phases of the project consist of: 1) the development of a central controlling program, 2) the development of instructional materials, and 3) the implementation of the system into the existing framework of the Office of Minority Student Education.

The MITE system uses a 48K Exidy Sorcerer microcomputer and the Exidy Display Disk Unit. The Display Disk Unit contains a 12 inch P31 video monitor and dual disk drives which accomodate single-sided, soft sectored, 5.25 inch disketts. The system operates under CP/M version 1.42/3 and all programs, with one exception, are written in Microsoft's MBASIC version 5.03. The single exception is written in Z-80 assembly language.

The block diagram shown in Figure 1 indicates the general structure of the central controlling program of the MITE system. This program has been written and essentially performs data processing operations. Each student authorized to use the system has his/her name and student number recorded on a data file along with other parameters relevant to MITE system operation. The user verification function is to ensure that the student is authorized to use the system and to establish various internal parameters used in the presentation of instructional materials.

One of the major goals of the MITE system is to provide instruction in a "tailored" form for the student using the system at the time. Using the parameters obtained from the verification process, the system will reconfigure the instructional unit to be presented into a form more suitable to the individuals needs and abilities. Once the unit has been "tailored" to the student it is processed and displayed on the video screen.

As the student works through the unit his/her answers to the questions and problems are recorded for later analysis. Once the entire unit has been presented, the student's responses are analyzed according to a series of grading algorithms obtained from a previously prepared data file. This file contains the answers to the questions in the instructional unit and several decision-making rules. A single unit may have up to five (5) separate grading algorithms depending upon the type of inferences to be made. Decisions regarding the sequencing of subsequent instruction and the level of complexity at which the instruction is to be presented to the student can be made. The MITE system finally records all of the decisions and student responses for later summative analysis.

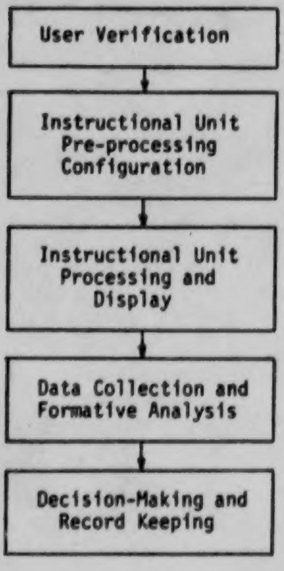

User Verification
Instructional Unit Pre-processing Configuration
Instructional Unit Processing and Display
Data Collection and Formative Analysis
Decision-Making and Record Keeping

Figure 1 - MITE System Operations

The remainder of this article presents a more detailed discussion of the MITE system data processing commands, the design of instructional units and various utility programs used in the preparation of instructional materials and data files.

### Instructional Unit Design:

The developer of instructional materials must perform two tasks when using the MITE system. The first is to write and prepare the instructional unit(s) in a form compatible with the MITE system data processing commands. The second is to prepare the decision-making algorithms for analyzing the student's responses. The strength of the MITE system lies in its ability to utilize the decision-making algorithms to link instructional units together. It is advantageous to construct each unit to be diagnostic where possible by preparing incorrect multiple-choice foils to reflect common errors in problem solving. Linking the incorrect foils to other instructional units via the decision-making algorithms leads to the effective remediation of misconceptions and misunderstandings.

The MITE system has been designed to utilize all of the features a computer has to offer. The system can be directed to assist in the construction of text materials, questions and problems. It can be directed to use its computing capabilities to draw graphs or to solve numerical problems. The system also has the ability to present instructional materials on three levels of difficulty. As mentioned in the introduction, the MITE system can tailor the instruction for the student using the system. This is accomplished by selecting questions of the appropriate complexity while ignoring those which may be too easy or hard for the individual using the system at the time. Instructional units designed and written to incorporate a differential level of presentation provide a more versatile and flexible means of instruction and thus address a wider, more diversified student population.

The functions listed above are performed through the use of special processing commands which will be discussed in greater detail later in the article. Most of the special processing commands focus on the construction of questions and problems, and it is through the questions and problems that the different levels of presentation are manifested. Each question or problem within the instructional unit can be classified according to one of three levels of difficulty (on a relative scale). Each student can also be assigned to one of three levels of unit presentation. The reconfiguration process, therefore, matches the level of questions or problems to be presented to the individuals assigned level. For example, a student who is assigned to receive easier, less complex questions will be presented with a subset of the unit's questions and/or problems which were classified as easy or less complex.

The system can process any instructional unit in a variety of ways. One way, already discussed, is to have the computer process the unit according to the individuals difficulty level assignment. If the instructor has not prepared the unit according to the difficulty level scheme or does not want the unit so presented, it is not necessary to do so. The instructor also has the option of deciding whether students will be allowed to page back through previously displayed material. In most cases, the instructional unit should permit the students to review text material and rework problems which were presented earlier in the unit. However, depending upon the content or purpose of the unit, students may be prohibited from paging back through the unit.

A third option deals with allowing students to use the computer as a calculator. The system has a built in multi-purpose calculator routine used to evaluate mathematical expressions supplied by either the instructional unit or the students. This routine was developed for the expressed purpose of eliminating the need of exiting the system during execution, performing one's calculations, and then continuing with the program. The break/calculate/continue situation is undesirable primarily because the system loses control of its own operation. Additionally, students using the system are not in a need to know position with respect to this type of computer operation. Again, depending upon the content or purpose of the unit the instructor may or may not permit the students to use the computer for calculation purposes. (This does not prohibit students from using hand held calculators or actually performing the break/calculate/continue sequence if they are aware of it.)

One final option concerning the presentation of instructional materials deals with special symbols or characters. The Exidy Sorcerer microcomputer has the capability of storing 64 user defined graphic characters (128 if the standard graphic characters are over written). The instructor has the option of selecting one of nine available user defined character sets. These character sets, each containing a maximum of 49 characters, are stored on a data file which is accessed by the MITE system upon execution. The MITE system and other support programs use 15 special characters for program control and are unavailable for general use.

The instructor also has several options available regarding how formative analysis of the unit is to be conducted. The MITE system can be directed to forego any form of analysis. Alternatively, the instructor may develop instructional materials requiring decisions concerning the sequencing of units where summative rather than formative analysis is appropriate. In this case the student's responses are analyzed but the student is only told which unit is assigned. Lastly, an instructional unit may be analyzed and the student informed of the results. The student's responses can be analyzed by a maximum of five different grading algorithms. These grading algorithms contain the responses for comparison and several decision rules. The responses for comparison are most often the correct answers to the questions and/or problems posed in the unit. However, if the instructional unit has been prepared for diagnostic use, the grading algorithms may contain incorrect responses along with decisions for assigning subsequent units to remedy the cause of the errors.

These various options compose the file parameter block (FPB) of the instructional unit and it is the first record of the unit. Its purpose is to establish controls over the presentation of the unit, set options available to the student and to prepare for the analysis of the student's answers. Since these parameters are part of the instructional unit itself, every unit prepared for the MITE system can take on "individual" or unique attributes and are not forced into some standardized format. Another benefit of this structure is in modifying any of the controlling parameters. Any changes in the presentation and/or analysis of a unit can be accomplished by the appropriate change in the first record of the unit's data file.

For those units to be reconfigured and presented according to the student's difficulty level, the instructor must prepare a "difficulty index block". This block of information is the second record of the unit and consists of the difficulty classifications for the questions and problems contained in the unit. Each unit can contain any number of questions but there is a limit of thirty questions which can be designated as scorable. The difficulty index block contains the relative difficulty classifications of the scorable questions. If the unit is to be presented according to difficulty level, questions are selected for presentation on the basis of the individual's assigned difficulty level and the item classifications in the difficulty index block. It is not mandatory that each unit be designed for differential levels of presentation. If the unit is not so constructed, then the difficulty index block does not have to be present in the unit.

Data Processing Commands:

The instructional unit consists of the file parameter block, difficulty index block (if necessary), and the textual materials. The text portion of the unit may contain whatever the instructor wants to teach. All of the information for a unit is stored and treated as a separate data file. The only restriction placed upon the text materials of the unit is that they conform to the data processing commands allowed by the MITE system. The MITE system responds to sixteen different data processing commands, which are listed in Figure 2. Each command is uniquely identified by five characters. Every line of text requiring some special process is prefaced by the five characters that define the process to be performed. These data processing commands actually become part of the unit's data file. Both the number of commands and their brevity will hopefully entice non-programmers into preparing instructional materials.

1. Input Command
2. Answer Command
3. Spacing Command
4. Question Command
5. Random String Command
6. Integer Number Generator
7. Decimal Number Generator
8. Text Phrase Generator
9. Compute Command
10. Compare Command
11. Foil Command
12. Select Command
13. Plot Command
14. Graphic Character Command
15. Assembly Language Routine
16. Mapping Command

Figure 2 - MITE Data Processing Commands

Any line of data (text) which does not require special processing is simply displayed on the video monitor. Therefore, it is possible to prepare a unit of straight text which is essentially copied from the input data file to the video display for the student to read. However, the system was designed to be more than an electronic page-turner. Through the use of these data processing commands the computer can assist in the construction of textual and/or testing materials.

1. Input command - This command is used to request the student's answer to a true-false or multiple-choice question. One other defining parameter of this command is the index of a data array for storing the student's answer. Answers that are stored will be analyzed and recorded upon completion of the unit. If the index is set to zero the student's answer is not stored in the data array. Therefore, it

is important in the designing of the unit to determine which questions to use for analysis and which to use for general practice and/or instructional purposes.

2. Answer Command - This command is used to provide immediate feedback to the student. The other defining parameter of this command is the correct answer to the multiple-choice or true-false question posed. The use of this command most often accompanies an Input Command in which the student's answer is not recorded for analysis. The computer will print an encouraging message for a correct answer or a consoling note and the correct answer in the event the student was wrong.
3. Spacing Command - This command was created to assist in the preparation of the textual materials, when those materials require vertical alignment. Before data is displayed on the video screen, the five left most characters of each line requiring special processing are removed. If several successive lines of data need to be vertically aligned and one or more of them require some kind of processing, text that was once aligned will appear offset on the video display due to the removal of five characters from the processed lines. This command provides five spaces of padding in order to preserve vertical alignments.
4. Question Command - This command is used to indicate the beginning of a question or problem statement. This command is important if the students are allowed to use the computer as a calculator. When a student enters the calculator mode of operation the question or problem statement may scroll off the top of the screen. This command is used to reprint the question once the student has returned from the calculator routine. This command is also used to indicate whether the question is scorable. Recall that if the instructional unit is to be presented according to difficulty level the computer selects questions for presentation based on their classification. Only the questions designated as scorable are subject to the selection process. One of the defining parameters of this command, therefore, indicates whether the question is scorable. The items that are not designated as scorable will be presented to the student regardless of the student's classification. The system also has the ability to generate textual information. The text generated is inserted directly into the line of data before being displayed. The data line that begins a question may have provisions for the generation of random variables (text or numbers). The last parameter used in defining this command is the number of random variables that are to be generated and inserted into the data line.
5. Random String Command - This command is used to indicate that random variables are required to complete the line of textual information prior to its display. The variables can be either text phrases or numbers and they are inserted directly into the text line which accompanies this command. The only other defining parameter is the number of variables to generate. The position in the text line where the generated variable is to be inserted is indicated by one of the user defined graphic characters that is reserved for system control. There are five different types of variables which can be used: integer numbers, decimal numbers, text phrases, calculated numeric values, or dependent text phrases. These five types of variables constitute several subroutines in the program that are called by the Random String Command. This command will read data from the input file, call the appropriate subroutine and replace the graphic character in the text line with the result. When all of the random variables have been inserted into the text line it is displayed on the screen.
6. Integer Number Generator - This command randomly generates an integer number. The number generated lies between minimum and maximum values which are supplied as the remaining parameters of the command.
7. Decimal Number Generator - This command is almost identical to the Integer Number Generator except that a divisor is included along with the minimum and maximum values. The command first generates an integer number between the minimum and maximum values, then divides the result by the divisor. The divisor is chosen to produce the desired number of decimal digits in preparing the number for display.
8. Text Phrase Generator - This command randomly selects text phrases from a list of possibilities. The command requires the number of text phrases available for selection. This parameter is used for the random selection of the text phrase. Text phrases can contain any number of characters. The command also accommodates any number of phrases. However, they must be contained on the single 64 character data line defined by this command.
9. Compute Command - This command is used to calculate a numerical value using numbers that were generated by the Integer and Decimal Number Generators. The result of the computation is inserted into the text line of the Random String Command which called this command. As each random variable is generated the system records the result in a data array. This allows the instructor to create questions which contain random numbers and still be able to ensure that the correct answer or other dependent numbers appear in the question statement. The result of the computation performed by this command is also saved. The other parameters used in defining this command are a number which specifies the number of decimal digits to retain in the answer and the mathematical expression used to calculate the answer itself.
10. Compare Command - This command is similar to the Input Command except that it is designed for numeric input. The parameters defining this command are the index for storing the student's answer and a mathematical expression to be used for comparing the student's input. The computer will ask the student for a numerical answer which it will compare to the result of the computation using the math expression supplied. If the two numbers are within .01 in absolute value, then the student is considered correct and the answer recorded is 'R'. If the two numbers differ by more than .01 in absolute value, then 'W' is recorded. As with the Input Command, the

answer does not have to be recorded for later analysis by setting the index to zero.

11. Foil Command - This command is used for the preparation of multiple-choice foils that are numeric in form. Each Foil Command prepares one line of displayed text consisting of a number computed from the mathematical expression that accompanies the command. The math expression usually involves numerical values that were randomly generated in the problem statement. The routine that processes this command can accommodate up to six foils and the foils can be displayed in one of three ways. The foils can be displayed in the same order that they are processed, or they may be randomized prior to their presentation. The third method of display is to present each foil one at a time. This method of presenting the foils requires the student to respond to each foil as it is presented. The student must decide and tell the computer whether the displayed foil is the correct answer to the question posed. If the student makes an incorrect response the question is terminated. If the correct answer is presented, the question is also terminated regardless of the student's response. If this option is used the foils will automatically be randomized before being displayed.
12. Select Command - This command is the textual counterpart of the Foil Command for it is used to select text phrases that are dependent upon other variables in the problem statement. The routine that processes this command can be called by the Random String Command to select, from a list, the text phrase that is linked to other variables in the problem which were randomly generated. The phrase that is selected would then be placed in the text line at the position indicated by the special graphic character. However, this command is most commonly used in the preparation of multiple-choice foils that are text phrases. In this context the Select Command is treated in the same manner as the Foil Command. Textual foils can be displayed via the three methods discussed in the previous section. The MITE system allows for both the Select and Foil Commands to be intermixed in the preparation of the foils. The actual structure of this command is the same as the Text Phrase Generator except that a mathematical expression is included in the data line. The value of this expression determines which phrase to use in the problem. There can be any number of phrases and they can be of any length, but they must be contained in the 64 character data line.
13. Plot Command - This command, as its name implies, directs the computer to plot a curve. The maximum X-value and the maximum Y-value are contained in the data line as is the math expression for the curve to be plotted. Any two-dimensional curve can be displayed on the screen. The curve is displayed in the middle 20 lines of the video monitor with 7 blank columns on each side, and the origin in the center of the screen. The computer performs the plot by calculating X-Y coordinate pairs and determining which point (pixel) on the screen needs to be turned on. The computer then constructs graphic characters and pokes them into the standard and user defined character spaces. The newly defined graphic symbols are then displayed on the screen. The plot cannot contain more than 128 characters since this is the maximum number of programmable characters. The system allows for the superposition of two or more curves as long as the 128 character limit is not reached.
14. Graphic Characters Command - This command is used to load an alternate set of user defined graphic characters into the computer's memory. This command can be used to reload the character set specified in the file parameter block after a graph has been plotted. It can also be used to load a different set of characters that are required by the instructional material that will follow this command.
15. Assembly Language Routine - This command is used when the instructor is unable to use any of the other commands to perform a desired task. This command provides 1500 bytes of memory for the loading and execution of an assembly language routine. The assembly language routine is accessed by the USR function of BASIC. The system permits only one assembly language routine to reside in memory at a time. But there is no limit on the number of routines that may be contained within an instructional unit.
16. Mapping Command - This command is not available to the instructor. It is an internal command used to unscramble the order of randomized multiple-choice foils. If the instructor opts to have the foils displayed in a random order, the system prepares a mapping algorithm which is used to translate the student's input response back into the original pre-scrambled order. This translation function is required so that the answers given by the student will correspond to the data contained in the grading algorithms referenced by the analysis routine.

These sixteen commands provide the instructor with a small but versatile set of data processing options. As with any author language, these commands require practice and experimentation. However, the fact that each command, excluding the Assembly Language Routine, is identified by a maximum of five characters facilitates the use of this system as a means of delivering instruction.

### MITE System Utility Programs:

The instructional system also contains a number of programs designed to assist in the development of instructional materials. The remainder of this article will provide a brief description of these programs and their uses.

Text Entry System - Once the instructional materials have been prepared in accordance with the data processing commands, the data has to be entered into the MITE system. An assembly language program has been prepared for the expressed purpose of providing a convenient and simple means of entering and storing instructional materials. The Text Entry System, accessed by two simple commands, gives the user complete control over the video screen. The keyboard acts just like a typewriter and all of the printing characters are available for use. The program was written because neither the BASIC interpreter nor word processing programs permitted the use of the standard or user defined graphic characters. Also, the user cannot type off the top or bottom of the video screen, and the system responds to all of the cursor controls. The data displayed on the video screen is permanently stored on a diskette and the screen cleared for a new page of data by using one of the control keys. The program allows for the creation, addition to, or editing of data files. The data files generated are sequential files that have excessive blanks removed for data storage optimization.

Unit Testing Program - Every unit prepared for the system is tested before it is fully implemented. A modified version of the data processing program is available to the instructors for performing dry runs of their materials. This program does not include the user verification or the data analysis sections of the larger system. The purpose of the program is to simply process the unit and display the instruction in the same form that the students would receive it. The instructor, therefore, can verify that each command is functioning in the way it was designed. Notes on the final form of presentation, construction of the questions and problems, or the aesthetics of the display can be made for later modification.

Character Generator Program - It is occasionally necessary to create special characters or symbols for the subject matter being prepared. Math and science areas usually have notations or symbols that are not available on the standard keyboard. This program was written to assist instructors in creating symbols that they require. The program displays an 8 by 8 grid representing the format for character construction. Through the use of various commands, the user can experiment with different combinations of light and dark squares in the grid. The symbol, in actual size, is also displayed. Once the user is satisfied with the symbol, the bit patterns that define the symbol can be stored into one of the user defined character sets.

Curve Generator Program - The process of specifying the parameters for plotting a curve are performed by this program. This program, like the character generator program, allows the user to experiment with various functions and parameters prior to their incorporation into an instructional unit. The program allows the user to input any function along with maximum X and Y limits. The computer will plot the function by using the same algorithm that is used in the data processing program. The utility of this program is to ensure that a plotted curve will not require any modification once it has been inserted into an instructional unit.

File Manipulation Program - The MITE system references a total of five random access files during execution. This program is a multi-purpose program that is used to create, edit, or delete data records from the random access files used by the data processing system. The five files are discussed below.

1. User Verification File - This file contains the student's name, student number, password and several data items used in the organization and presentation of the instructional materials. The file manipulation program is used to create data records for each student receiving authorization to use the MITE system. As the student completes instructional units, his/her data record is updated by the data processing system. The file manipulation program can be used to either view, or change the student's status.
2. Grading Data File - This file contains the grading algorithms for each unit in the MITE system. When an instructional unit has been presented to the student, his/her answers to the questions posed in the unit can be analyzed by up to five different grading algorithms. These algorithms are entered into this file by the file manipulation program. Each algorithm contains the responses to be compared with the student's answers and several decision rules. The decision rules determine the assignment of subsequent instruction and the level at which the instruction is to be presented.
3. Record Keeping File - The final task of the data processing program is to store the student's answers, the scores generated by the grading algorithms, and any decisions made. The file manipulation program is used only for the purpose of viewing this data. It can be used to modify information stored in the record keeping file, but this is rarely done.
4. Comments Data File - At the completion of an instructional unit the student is asked for any comments s/he might have. These comments are recorded on this data file in free format form. Each comment is prefaced with the instructional unit's name for reference. The file manipulation program is useful in listing these comments.
5. Symbol Data File - As discussed earlier, the MITE system has provisions for storing user defined characters or symbols. This data file contains the nine different character sets that can be accessed by the data processing program. The file manipulation program is used to create data records for this file.

Summative Analysis Program - This program is used to summarize all of the data pertaining to the same instructional unit. At the end of the term the instructor can perform several different types of summative analysis. This program accesses the record keeping data file and can perform analysis on a particular unit, a particular student or even a group of students. The underlying purpose of this program is to provide insights into the usage and effectiveness of the MITE system as a method for delivering instruction. The types of analysis that can be performed also provide guidelines for the modification of instructional units.

Each of these programs has been designed and written with the user in mind. The access procedures for each of these programs is performed by typing two simple commands. The programs themselves require a minimum of input from the user, making them simple and less threatening to use. The entire MITE system does not require any prior computer programming skills to operate. In this way instructors, tutors, and in-class aides are given the opportunity to prepare instructional materials for others.

Summary:

This paper has presented a description of the Microcomputerized Instructional, Testing and Evaluation system being developed at Michigan State University. The system consists of five major components: a data processing program, a text entry system, a set of random access data files, several utility programs, and the instructional units.

The basis for developing this system was to provide students with a supplementary source of text information, drill and practice exercises, or even test questions. The content of the instructional units is up to the discretion of the instructor or curriculum developer. The only constraint imposed by the system is that the text lines in the unit's data file conform to the data processing commands allowed. The sixteen commands permitted establish a small yet versatile set of data processing options.

The data processing system not only processes the information contained in the unit, but it supplies the interface with the students. The system also performs analysis and evaluation of the student's responses to questions posed in a unit. The results of the analysis can then be used to assign subsequent instruction to the student.

It is through the integration of the instruction, testing, and analysis processes that effective and efficient education can take place. This system requires the conscientious design and unification of these three aspects of education in the preparation of instructional materials.

# A STRUCTURED APPROACH TO THE DEVELOPMENT OF COMPUTER ASSISTED INSTRUCTION

JANET HARTMAN - DIANNE GARNETT
UNIVERSITY OF EVANSVILLE
P. O. BOX 329
EVANSVILLE, INDIANA 47702

## INTRODUCTION

The use of the computer as an instructional tool is an effective way to provide for individualized needs, improve the availability of classroom resources, and motivate students. Computer assisted instruction (CAI) helps the instructor to optimize his use of time by freeing him of routine tasks like grading papers and presenting repetitious material such as drill and practice. It can also enhance the instructor's potential by assisting in the diagnosis of learning problems and the evaluation of student progress. The decreasing cost of computer hardware and the advent of the microcomputer have made CAI available in many classrooms which previously could not afford the expense of time sharing. Future teachers from the elementary school level up must be aware of the instructional capabilities of the computer. Many may, in fact, develop materials for use in their classrooms.

A successful CAI module or course depends on a structured top-down design which begins with a clear, concise educational goal and analysis of audience characteristics. The process follows further refinement through the development of specific educational objectives and the selection of an educational strategy before beginning to write content or programming code. In this paper a structured approach to CAI development, including assessment of course goals and learner characteristics, task analysis, selection of an instructional strategy, organization of content, authoring techniques, module or course testing and evaluation of the finished product, will be addressed.

## DESIGN PHASE

The design of CAI requires a system approach. This approach is independent of the instructional strategy chosen whether it be a test, tutorial package or simulation. Good CAI material should be carefully planned with firm curricular goals in mind and based on sound principles of learning. The process of conceiving and designing the instructional content of CAI involves the six basic steps outlined below:

### Step 1: Assessment of Course Goals and Learner Characteristics

Before starting to write any CAI materials, the instructor should review the goals of his course in order to determine how specific content can be best presented. The method by which goals have previously been attained should be evaluated. If goals can be met with more efficient management of time and resources or if learning can be enhanced through the use of CAI, then the decision to proceed with the design of a specific CAI materials is valid.

In addition to considering course goals the particular characteristics of the students who will be using the CAI materials should be considered. What are their needs? Their interests? What is their reading level? Their level of intellectual development? How ready are they for experiencing the material? The computer as a classroom tool? Can CAI provide for individual needs or improve the availability of instructional materials? Will CAI ultimately enhance the quality of learning which students will experience?

Evaluation of course goals and learner needs are essential to well-planned instruction. Once needs have been identified an instructional strategy and a topic can be chosen, and the content can be organized into a meaningful learning activity.

### Step 2: Selection of a Topic

The topic chosen for CAI materials depends on learner needs, course goals, teacher interest and adaptability to the computer. Since CAI requires a lot of time to develop, it is wise to narrow the scope of the content into a manageable unit. For an initial attempt at developing CAI materials, choosing a topic for which students will spend one-half to one hour at the computer will be manageable.

It is helpful to identify specific tasks or content areas which are subordinate to the topic chosen and order them. These areas should be as simple as possible--focusing on one task to be performed or one concept to be learned. Identification of subordinate areas facilitates the establishment of learning hierarchies and related specific objectives.

### Step 3: Selection of An Instructional Strategy

In a limited sense the computer can be programmed to provide drill and practice, function as a tutor, pose problems, and administer exams. It can also be programmed to record and store data for individual students from any one of these strategies. A combination of these strategies may also be used. For example, a tutorial package might include a pretest, textual development, drill and practice and a posttest. Some activities which may be developed include questions and answers, drill and practice, problem solving, case studies, and simulations. All require interaction of the student with the computer, but the amount of interaction may vary with the content.

### Step 4: Establishment of Learning Hierarchies

From the subordinate areas which were identified in Step 2, the task or the component of the content to be considered should be stated. Each task or content area should be subjected to a task analysis to identify the independent and dependent sequences which lead to performance of the task or acquisition of the content. Another outcome of the analysis is the identification of prerequisite knowledge and skills which are essential to performance of the task or mastering the content. The most efficient way to identify these learning hierarchies is to use Gagne's method of task analysis. In this process content is subdivided into descending levels of cognitive products and processes from abstract or general to concrete or simpler. The terminal task is stated and skills or knowledge which are prerequisite for accomplishing the task are arranged in a hierarchial format (see Figure 1). The various levels identified in this hierarchy facilitate the statement of specific objectives and construction of learning activities for fulfilling them.

Once all of the tasks analyses have been completed the subordinate areas can be sequenced and a basic outline of the content to be presented is in order. At that point, there remains one additional consideration before proceeding to writing specific objectives for the tasks or content in the learning hierarchy. The task analysis has ordered content from basic facts to general principles, but the instructor must also decide whether to give the student a series of facts and lead him to develop the principle (inductive presentation) or present the principle first and have the student substantiate it with facts (deductive presentation). Whether the material is to be presented inductively or deductively, a task analysis organizes the procession of content. Since the topic being presented is likely to have several subordinate areas, it may be possible to exercise either one or both options.

### Step 5: Statement of Specific Objectives

Specific objectives are primarily derived from the individual task analyses conducted in the previous step. These should be behaviorally stated, identifying observable skills or performances expected of the learner following

## TASK ANALYSIS

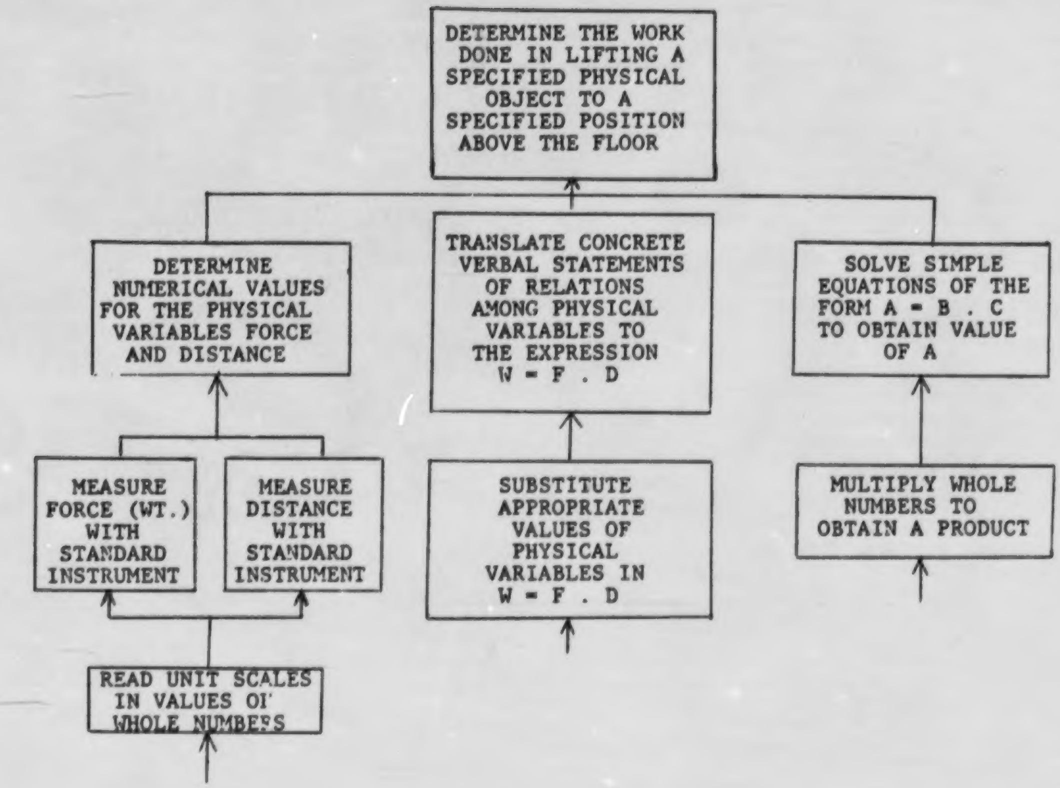

DETERMINE THE WORK DONE IN LIFTING A SPECIFIED PHYSICAL OBJECT TO A SPECIFIED POSITION ABOVE THE FLOOR
DETERMINE NUMERICAL VALUES FOR THE PHYSICAL VARIABLES FORCE AND DISTANCE
TRANSLATE CONCRETE VERBAL STATEMENTS OF RELATIONS AMONG PHYSICAL VARIABLES TO THE EXPRESSION W = F . D
SOLVE SIMPLE EQUATIONS OF THE FORM A = B . C TO OBTAIN VALUE OF A
MEASURE FORCE (WT.) WITH STANDARD INSTRUMENT
MEASURE DISTANCE WITH STANDARD INSTRUMENT
SUBSTITUTE APPROPRIATE VALUES OF PHYSICAL VARIABLES IN W = F . D
MULTIPLY WHOLE NUMBERS TO OBTAIN A PRODUCT
READ UNIT SCALES IN VALUES OF WHOLE NUMBERS

FIGURE 1

instruction. The outcome must be observable so both instructor and student will know when the objective has been met. Objectives should be adjusted to reflect consideration of learner needs.

Since CAI is designed to enhance the cognitive growth of the student, various levels of the cognitive thinking process should be reflected in the statement of objectives. The most widely used classification system for analyzing objectives in the area of cognitive processes was developed by Bloom. In Bloom's taxonomy the cognitive processes are arranged in hierarchial levels from simple fact retrieval to complex evaluation based on decision-making principles (see Figure 2). It is desirable to include objectives which require higher order thinking (comparing, analyzing, evaluating) as well as objectives requiring simple fact retrievel.

While the cognitive domain is the primary target of learning, the instructor should also be aware of the affective domain. Specific attitudes and feelings related to course content or use of the computer should not be dismissed as unimportant. Students may emerge from the experience with CAI being more motivated. This in itself is a worthy accomplishment. On the other hand, students may be anxious or feel the method to be impersonal; this feeling too, must be recognized.

Step 6: Organizing the Content Framework

Using the sequence of subordinate areas identified in Step 2 and the objectives listed in Step 5, a content framework can be constructed. In this framework the objectives should be written in an appropriate sequence, this facilitates the writing of specific items or questions in the text of the CAI materials. At this point decisions about associated learning resources can also be made--student guide, readings, etc.

The design for the CAI materials is now complete; the next state is writing the text. Before proceeding to the writing stage, it may be helpful to ask knowledgeable colleagues to validate the content framework for format, sequencing and context. Procedural omissions, errs, and ambiguities can be resolved easier at this point than after the material is on the computer.

## WRITING PHASE

If the particular capabilities and idiosyncrasies of the computer for which the course material is being written are unfamiliar, consultation with computer personnel would be beneficial at this point. Questions about branching, item pools, and other procedural technicalities which may make the material and programming more effective or efficient can be answered before the job of writing the text and items is underway. It may also be possible at this time to determine who will be doing the actual programming of the material, if you are not doing it, and to make the initial contact with the programmer.

The presentation of the text material and items to which the user responds must be clear and free from ambiguity. Initial information should include a discription of specific machine features, the mode of response to be used, an estimate of time needed to complete the material, definitions of abbreviations and special terms, and any other information helpful to successful completion of the material. Instructions should be clear, simple, and prominent. They should not be buried in the text. The selection and use of one format for presenting instructions and messages will make them easier for the user to recognize and for you to develop.

Clarity can be enhanced further by the effective use of headings, appropriate writing style, summaries and transitional text. Headings should appear at the beginning of each topic. A summary at the end of a topic will provide reinforcement of the concepts presented and a sense of closure. Sentences should be short, concise, and reflect the vocabulary level of the user. Key words can be emphasized through the use of a brighter light intensity and they should be marked by an asterisk or underlining in the written material to be programmed. Long sections of text should be avoided, using short paragraphs interspersed with student interaction instead. This helps to keep the student interested and involved. Transitional text will help to optimize the flow of the program. Effective transitions can be arrived at by utilizing such phrases as, "Let me offer an example..." "By the way, that reminds me...", etc.

The impersonal nature of the computer can be somewhat diminished by calling the student by name and by allowing the personality of the author to emerge in the text material. Insertion of the student's name, humor, and conversation in the text of feedback can remove the machine from between the instructor and the learner. Branching can also help to personalize learning for the user by providing him with various options as to the direction or sequence which he may utilize within the program. Each student can potentially have his own personally prescribed program of learning.

In providing responses to student input, a provision for alternate answers, should be made which is as flexible as possible as to what will be accepted.

## LEVELS OF COGNITIVE THINKING*

| Knowledge | Comprehension | Application | Analysis | Synthesis | Evaluation |
|---|---|---|---|---|---|
| (ability to recall; to being to mind the appropriate material | (ability to comprehend what is being communicated and make use of the idea without relating it to other ideas or material or seeing fullest meaning) | (ability to use ideas, principles, theories in new particular and concentrated situations) | (ability to break down a communication into constituent parts in order to make organization of the whole clear) | (ability to put together parts and elements into a unified organization or whole | (ability to judge the value of ideas, procedures, method using appropriate criteria) |
| | | | | | Requires synthesis |
| | | | | Require analysis | Requires analysis |
| | | | Requires application | Requires application | Requires application |
| | | Requires comprehension | Requires comprehension | Requires comprehension | Requires comprehension |
| | Requires knowledge | Requires knowledge | Requires knowledge | Requires knowledge | Requires knowledge |

FIGURE 2

---

*Wiles and Bondi. Curriculum Development A Guide to Practice, 1979.

For example, if "true" or "false" is requested, "T" and "F" should also be allowed. Unanticipated responses or spelling (unless correct is desired) can be provided for by including one or more unanticipated response statements. This will accommodate the student who thinks of something which the author didn't or types in "help". The student should be made aware of when he is to respond. For example, the student should be told to enter the correct letter, not just given a list and expected to respond. For example, the student should be told to enter the correct letter, not just given a list and expected to respond. Short, simple responses given one at a time should be requested. Having the student enter several answers at the same time, creates problems in evaluation and is confusing. Consistency in the type of response requested will also help to eliminate confusion.

In providing feedback to the student, each of his responses should be acknowledged, including relevant replies to anticipated wrong answers. If possible, remedial material to help the student who responds incorrectly should be provided to help him determine the correct answer. Feedback to students should be given in a positive manner and replies should be short, simple and varied from question to question. Use of the student's name in the feedback occasionally is also desirable.

## EVALUATION PHASE

Once the text has been completed and the materials have been programmed for the computer, the material should be evaluated at the terminal to determine its strengths and weaknesses. It would be beneficial to have other reviewers evaluate the course materials also. For consistency in evaluation each reviewer should be provided with a list of checkpoints which the author considers pertinent to the effectiveness of the program. After a critical review initial editing can be done. Additional editing will probably be necessary after students have used the materials in an instructional situation. If a more formal evaluation is desired, statistical measures relating to item difficulty, item discrimination, and reliability could be used.

## CONCLUSION

Learning can be accomplished in a variety of ways--CAI being only one. The learning environment varies according to learner characteristics, type and variety of content and other variables. With CAI the environment can be more flexible, objectives are more explicit, the learner is an active participant, learning can be more individualized, evaluation and diagnosis are more efficient and the teacher's role is more facilitative. These advantages of using CAI make the initial time invested in development worthwhile.

## REFERENCES

Burson, Jeanne L. The Author's Guide to CAI. Columbus, Ohio, Division of Computing Services for Medical Education and Research, Ohio State University, 1976.

Casciano-Savignano, C. Jennie. Systems Approach to Curriculum and Instructional Improvement. Columbus, Ohio: Charles E. Merrill Publishing Co., 1978.

Dimas, Chris. "A Strategy for Developing CAI," Educational Technology, 18 (April, 1978), 26-28.

Edwards, Judith et al. Computer Applications in Instruction. Hanover, New Hampshire: Houghton Mifflin, 1978.

Wiles, Jon and Bondi, Joseph, Jr. Curriculum Development A Guide to Practice. Columbus, Ohio: Charles E. Merrill Publishing Company, 1979.

# A PARADIGM FOR DEVELOPING A COMPUTER SCIENCE PROGRAM

Johnette Hassell, Department of Computer Science
Tulane University, New Orleans, Louisiana 70118

## Abstract

Many schools across the country are contemplating the creation of a computer science program. The current shortage of computer science graduates makes it difficult to build a program by hiring new faculty. This paper presents a paradigm for developing a computer science program. This paradigm is incremetal, makes use of present faculty, and allows the program to be stabilized at the discretion of the college or university.

## INTRODUCTION

Many schools are faced with the necessity of developing, from the ground up, a Department of Computer Science. This paper addresses the problems arising with the creation of a Computer Science Department and presents a paradigm for an incremental approach to its development. While many of the problems discussed here typically apply to small, private or junior colleges, they may apply to any insitution without a fully functional Computer Science Department.

The need for a computer science curriculum grows from many sources. As new faculty bring with them knowledge of computer usage in solving the problems of their discipline, they require extensive hardware and software resources. Understandably, they wish to use these computer techniques in teaching. Thus they require that students have computer experience. In many cases, the need is met by the physicist, mathematician or social scientist who has learned to program. He/she offers some introductory applications of the discipline involved. Soon however, these "programming" courses are populated by students from other disciplines. If nothing interferes, the faculty member involved becomes the campus "programming expert" and must divide his/her time between the computer courses and the original field of interest.

An alternate senario finds faculty in each of several different departments teaching programming to their own students. This leads to the obvious waste of duplication of time, resources, effort and personnel. It is frequently the case, also, that issues of territorial authority arise. In either case, the student is not presented with a coherent, comprehensive view of computer science, computer usage or even good programming.

Pressures to offer computer science courses also come from students themselves. Computing has a reputation for being a glamorous, highly-paid career field. This attracts students to Computer Science and consequently to schools with such programs. Thus schools without computer science offerings miss out on some of their brightest prospective students.

As computers have decreased in size and cost, they have become available to students in high school. Thus many students arrive at college with some knowledge of computing. They are discouraged when they find that the school offers little more than that of their previous high school. Again, students expect a functional computer science program and will choose a college or university accordingly.

A student does not always enter college with an exact career field in mind. Once in a college or university, a student may wish to switch to computing. If no program beyond elementary programming is available, he/she is faced with two alternatives: transfer elsewhere or exert pressure on the institution to create a computer science program.

Many school administrations would like nothing better than to create a Computer Science Department. But they are faced with serious, and virtually insurmountable, problems. These problems fall into two categories- financial and personnel. Surprisingly to those familiar with the ongoing financial problems of education, the personnel issue is the more difficult one.

There simply are not enough computer science graduates to go around. Each year fewer Ph.D.s in computer science are produced [3,4]. Industry insists on offering students, with only bachelors degrees, salaries and job opportunities that are so attractive that further formal education seems counterproductive. Students are told plainly, in fact and in action, "Don't go to graduate schoool. We'll teach you all you need to know." This continued short-sightedness on the part of industry is a major contribution to the lack of computer science Ph.D.s.

Those new Ph.D.s with the desire to go into academics are drawn to established computer science departments with active research programs. Rarely are they willing to go to a school which is in the process of developing a computer science program.

Other problems plague institutions as they try to build new departments. In recent years, many schools have become "top heavy" with as much as 90% of the faculty having tenure. The trustees of some institutuions have resorted to setting a maximum overall tenure rate for the faculty. Some state schools have maximum salaries that are far lower than is commensurate with the job. Low salaries and reduced access to tenure further hamper efforts to hire computer science faculty.

Such problems are part of a national trend of reduced faculty mobility. This poses very real and personal problems for some faculty. After years of schooling and more years of teaching, research, curriculum development, and committee work, some faculty lose interest in their work and begin to question their career choices. This feeling has been dubbed the academic mid-life crisis and is a rather common occurence. In this author's experience, such feelings often occur as a result of a need to recover the feeling of challenge and achievement that characterized successful graduate days. Yet few faculty can afford to quit teaching and return to graduate schoool for "retreading."

These problems seem to be diverse and unrelated. Yet in seeking a solution to one, others may be partially, if not completely, solved. The following section advocates a paradigm for developing a academically sound computer science program that makes use of current faculty who are ready to conquer new worlds.

## DEVELOPMENT OF A COMPUTER SCIENCE PROGRAM

In this section we hypothesize about University X and show how it can follow this paradigm to develop a computer science program. These procedures have been and are being used. The University described here is fictious, but typical.

The University of X is a private institution with a strong Department of Mathematics and an active pre-engineering program tied in with a larger state school. Faculty in both the Math and Physics Departments have experience with programming and are teaching courses in FORTRAN for their majors. These two faculty are interested in computer science and in having a computer science program, but have little knowledge of the area beyond programming applications. The University would like to have a Computer Science Department, but has not been able to attract a suitable faculty member to head it. Furthermore, the Board of Trustees is skeptical about creating an entirely new department and degree program until it is know that a sufficient number of students will enroll. Of course, one cannot demonstrate that a sufficient number of sudents are available without a program in place. Nevertheless, the Board is willing to experiment with a computer science program with the following provisions:

1. Present faculty should be used if possible;

2. The program should be installed incrementally;

3. The program should be academically sound at all stages of its development;

4. If at any time it is determined that no further expansion is desired, the program development can be stabilized;

5. The program should not require large "front end" expenditures for faculty.

The keystone to meeting these objectives is the use and development of a course called Foundations of Computer Science. It is a survey course that includes an number of topics - data structures, formal languages, graph theory, and boolean algebras- all of which are related to discrete mathematics [2]. With a moderate amount of reading [1,7,8,9], access to computer resources, and support to attend conferences, our hypothetical faculty members can develop and teach this course.

The course must be strongly project-based; that is, it must require a number of moderately complex programming assignments. A most appropriate project is the development of an intepreter for a simple algebraic language. Such a project gives students a better, more comprehensive view of a computer system. The theoretical nature of the subject matter is a vital force in a students achieving maturity.

Our hypothetical faculty members probably learned programming by the "seat of the pants" method and have little concept of the necessity for software design and its techniques. Experience has shown that managing students in projects of this complexity quickly emphasizes the need for the tools of software design. Developing such a course also helps the would-be computer scientist get a view of the computer science world that extends beyond programming.

The Foundations course should be taught for at least 2 or 3 years, although it can be taught indefinitely if it meets the needs of the University. During this time, the University can assess the need for, and popularity of, the Computer Science program. This "breathing space" is invaluable to the University that must make difficult decisions about continued expansion of the program. A student in the program at this stage is getting a sound, broad view of computer science. He/she is prepared to enter a training program in a computer industry or qualify as an entry level programmer. For the faculty involved, it is a time of tremendous growth and discovery of an exciting new world.

If the University decides to expand the program, a basic pattern is followed. One of the topics in the Foundations course is "spun off" into a full fledged course of its own. The resulting space in Foundations is filled with new topics or greater coverage of topics already included.

The first spin-off should be a Data Structures course. Data Structures should follow the introductory programming courses. There are a sufficient number of texts available to provide adequate support for this course [10]. Foundations should be moved to a junior level status with topics on the analysis of algorithms to replace the data structure material.

If assembly language has not been offered, it too must be added to the curriculum at some point. Typically adding an elementary assembly language course is not too difficult and personnel from the computer center may be used to teach it.

At this point in the program development, the University can offer a concentration or emphasis in Computer Science to accompany suitable majors in other fields. Also, at this point, the faculty should be involved in reading the research literature in the basic areas covered in Foundations. Support in the form of memberships to ACM and its various special interest groups and the IEEE Compuer Society should be provided. Each of these organizations offers working conferences or short courses with in-depth technical papers. Travel to selected meetings are a economic way of broading the horizions of the faculty.

It is also appropriate, at this time, to consider a fully paid year of work for the faculty at a University with an active Computer Science Department. This author's experience indicates that summers alone or the usual one semester sabbatical leave are insufficient. When both the relative unavailability of computer science Ph.D.s and the salary they command are considered, one year of paid leave for faculty seems like a small investment. Furthermore, external funding is often available for such study.

For faculty who have become bored, stagnant or who are facing an academic mid-life crisis, developing and teaching the Foundation and Data Structure courses can prove very stimulating. With the kinds of support described, these faculty may become active, practicing computer scientists. Certainly the kinds of support advocated here provide a school administration with a powerful source of motivation for faculty development.

The next step in the paradigm is to collect the topics in Foundations that relate to software design and design methodology, and create from them a Software Engineering course. The space thus created in the Foundations course should be filled with topics on computer architecture. At this point, the University can offer a Computer Science minor. This will prove to be a natural stopping place for many schools. Certainly the University should offer the Computer Science minor for two to five years while additional courses such as Operating Systems and Computer Architecture are spun off from the Foundations course. This provides more flexibility and allows students some choices in their Computer Science coursework.

To offer a Computer Science major requires considerably more course offerings and faculty. If the University wishes to create a major program, several things can be done to facilitate it. One must remember that new faculty will only be attracted to established departments with active research potential. Therefore the Computer Science faculty should be encouraged/allowed to develop a variety of new courses to indicate they intend to offer a major program.

If the current faculty are beginning to produce computer science research, such activities will be of great help in attracting new faculty. If not, current faculty should be given extensive support as they begin their research efforts. This may involve summer support, on-going consultants, release time, library funds, as well as the funding for professional activities advocated earlier.

Many computer-based industries have faculty loan programs. They should be investigated and utilized whenever possible. Visiting professorships, at attractive salaries, can be used to bring in computer scientists of known reputation. This provides intellectual stimulation to permanent faculty and increases the department's versatility. Faculty with sabbatical leave from other institutions also provide a source of temporary assistance. While each of these ideas provides only temporary faculty, they are invaluable for establishing contacts, reputation and new ideas.

It is important to note that this plan provides an opportunity for a faculty member to become a practicing computer scientist and to extend his/her knowledge beyond the narrow view of programming. Computer Science should be taught by computer scientists just as history should be taught by historians. This paradigm allows interested faculty to grow as computer scientists as the program itself grows.

## CONCLUSION

The paradigm presented here allows for a planned, incremental development of a Computer Science program. It takes advantage of available faculty with known teaching abilities and loyalty to the institution. The incremental nature of the program allows for faculty to grow and develop as professional computer scientists. It also allows the university to maintain control over the program and terminate the development stage when an appropriate level is reached.

## BIBLIOGRAPHY

[1] Berztiss, A.T., Data Structures, Second Edition, Academic Press, 1975.

[2] Benard, M., "A Foundations Course for a Developing Computer Science Program," SIGCSE Twelth Technical Symposium on Computer Science Education, SIGCSE Bulletin, vol. 13, No. 1.

[3] Conte, S.D. and O.E. Taulbee, *Production and Employment of Ph.D.s in Computer Science*, University of Pittsburg, 1978.

[4] Hamblen, J.W., *Computer Manpower Supply and Demand*, Information Systems Consultants, St. James, Missouri, 1979.

[5] Hamblen, J.W., "Computer Science Ph.D. "Glut" Nonexistant," COMPUTERWORLD, April 1, 1979, p. 24.

[6] Knuth, D.E., *The Art of Computer Programming*, 3 volumes, Addison- Wesley.

[7] Korfhage, R.R., *Discrete Computational Structures*, Academic Press, 1974.

[8] Marshall, C.W., *Applied Graph Theory*, Wiley-Interscience, 1971.

[9] Tremblay, J.P. and R. Manohar, *Discrete Mathematical Structures with Applications to Computer Science*, McGraw-Hill, 1975.

[10] Tremblay, J.P. and P.G. Sorenson, *An Introduction to Data Structures with Applications*, McGraw-Hill, 1976.

# MANAGEMENT APPLICATIONS OF THE MICROCOMPUTER: PROMISES AND PITFALLS

Dr. John E. Haugo
President
EduSystems, Inc.
St. Paul, Minnesota

ABSTRACT: The rapid advances in microcomputers in computing has made many educational applications feasible which formerly were performed on large scale systems. Initial applications have been primarily instructional in nature. This paper will deal with potential management applications of the microcomputer. Applications which relate to financial, personnel, and student information systems will be discussed. Primary emphasis will be on the relative advantages and disadvantages compared to large computer systems and manual record-keeping processes. Alternatives for the development and support of microcomputer-based management systems will also be discussed. School district administrators and data processing personnel will find this session of interest. As educational decision-makers look at means to cut costs, yet increase management productivity, they should consider potential microcomputer-based applications.

## Background

The availability of a variety of microcomputers over the past several years has created a great deal of excitement among educators in the United States. The new computers have offered an opportunity to learn new aspects of computing at a low cost. As the microcomputer has begun to be used by persons who work closely with users of more traditional educational computing services, these persons also began to develop programs to show selected capabilities of the new devices. The use of microcomputers in school districts has led to an increased awareness among many school district administrators of the processing capabilities of computers. Particularly in small districts and in individual schools in large districts, interest has been expressed in using microcomputers to handle some administrative activities.

As educational administrators have gained some familiarity with microcomputers, opinions as to the potential value of computers for management tasks have been more prevalent, and more extreme. Claims have been made at both ends of the continuum. Some say that microcomputers are capable of handling all of the management computing needs of small districts and schools, while others say that they are appropriate only for instructional computing needs. However, most educators who have an awareness of the microcomputers see potential for their usage as management tools and are asking for more information about applications that might be feasible.

In the private sector, the use of microcomputers for administrative purposes has increased significantly during the past year. Microcomputer company sales and projections show that small business use of microcomputers is going to be an increasingly large portion of the market. Tandy, Apple, and Commodore have all recently enhanced their microcomputer configurations to accommodate small business data processing applications. During the past several years, literally hundreds of microcomputer-based applications such as general ledger, accounts/receivable, accounts/payable, and inventory applications have been developed for the small business user.

During the past year, the Minnesota Educational Computing Consortium (MECC) has explored several approaches in the use of the microcomputer as a management tool. First, they have developed the interface between a large host computer and an Apple II microcomputer. The Apple II is used as a remote data collection device which edits, stores, and transmits the data to the host computer. This approach reduces the load on the host and on the telecommunications network. MECC has also worked with several school districts on the development of stand-alone management applications.

The 1981 Apple Software Directory published by Manlove includes a section on school administration applications. It describes twenty-two available applications for the Apple II. Attendance, class scheduling, school bus routing, fixed asset inventories, and central activity accounting are among those listed. Prices vary from less than $20 to over $400 per application.

A number of publishing companies have announced microcomputer-based school management applications. The most common application is a course on instructional management systems which relates to a particular textbook series. Recently, Scott, Foresman and Company announced a comprehensive set of school management applications. The applications which are developed to operate on the Texas Instrument 99/4 microcomputer related to a variety of student, personnel, and financial functions. The fifteen applications are grouped or packaged to be utilized at the district, school, or classroom level.

## Feasibility of Microcomputer-Based Management Applications

During 1979, there were two efforts to determine the feasibility and/or interest in school management applications of the microcomputer. Education Turnkey Systems, Inc. of Washington, D.C. through its Microcomputer Education Application Network (MEAN) surveyed educational administrators at several national conventions. They determined that there is a high level of interest in educational and administrative applications of the microcomputer. The preliminary findings of their survey showed that there was significant interest in a number of applications including:

- Monitoring of Individualized Special Education Programs
- Test Scoring and Analysis Programs
- Title I Report Generator
- Computerized Curriculum Guides
- Student Scheduling
- Attendance
- Equipment/Materials Inventories
- Grade Reporting

The second study of feasibility and interest in administrative users of the micrcomputer was conducted by the Minnesota Educational Computing Consortium. Because of the widespread availability of Apple II's in school districts in Minnesota, there was interest in using them for management purposes.

The MECC study team used a variety of data gathering techniques including site visits at school districts who were using the microcomputer for school management purposes, usage data gathered by regional MECC coordinators, interviews with school superintendents, visits with small business microcomputer users and computer vendors, and a thorough analysis of existing timesharing or microcomputer administrative software. General findings regarding the technical feasibility were:

Processor Speed. Compared to large computers, microcomputers are slow and can handle relatively small amounts of data. However, for most of the potential applications explored in the study, speed is not critical. Also, for many applications (such as payroll) the number of records needed would be in the range of 100-200. For most potential applications, processor speed would seem to be adequate.

Memory. It is estimated that for most of the potential applications, 32K of random access memory internal to the computer would be adequate. The memory could be expanded to 48K with the purchase of 16K of memory for a relatively small price. The additional memory would decrease processing time for some applications, such as those requiring extensive sorting.

Input. Several input modes are possible for the microcomputer. A user may utilize the keyboard, punched cards, mark sense cards, cassette tape, diskette, or a downloading connection from another computer system.

Disk Storage. Multiple disk drives can be connected to most microcomputers. Each disk drive allows storage to up to 116,000 characters that can be retrieved through random access. Up to 1200 records of 80 column length can be stored on one diskette.

Output. As with input, several output modes are possible with microcomputers. A user may utilize a hard copy printer, a television screen, a video monitor, a disk drive, or a cassette tape. Output speeds varying from 30 characters per second (cps) to 120 cps are possible.

Reliability. The reliability of microcomputers is at this point an unknown. While we have no indication of major problems which would cause the frequent loss of data, occasional problems could be expected to occur. Back-up equipment and procedures would considerably reduce the risk of data loss. At this time, microcomputers have not been around long enough to adequately assess reliability.

## Potential Applications

In determining a list of possible microcomputer-based school management applications, it is necessary to examine those which currently are available; to identify applications used in the business sector which have applicability for education; and to dicuss potential applications with persons who have expertise in both school management and computer technology.

As a result of this apporach and based on the aforementioned technical feasibility considerations, the following is a list of potential applications by area:

Student:

1. Atheltic Eligibility List
2. Attendance (annual)
3. Attendance (daily)
4. Class Records
5. Census (family)
6. Enrollment Projection
7. Graduate Follow-Up
8. Guidance Records
9. Health Records
10. Instructional Management
11. Mark Reporting
12. Scheduling Assistance
13. School Calendar
14. Student Records
15. Test Scoring and Analysis

**Personnel:**

1. Paycheck Calculation
2. Payroll Reporting
3. Personnel Record
4. Salary Simulation
5. Staff Assignments

**Facilities:**

1. Energy Management
2. Facilities/Equipment Inventory
3. Facilities Utilization
4. Maintenance

**Finance:**

1. Accounts Receivable/Payable
2. Activity Accounting
3. Financial Forecasting
4. Food Service
5. General Accounting
6. General Ledger
7. Investment Accounting
8. Vendor Reports/Purchase Orders

**General:**

1. Activity Scheduling
2. Ad Hoc Reporting
3. Bus Routing
4. Information Storage and Retrieval
5. Library Circulation
6. Media Reservations
7. Mailing Lists/Labels
8. Project Planning and Budgeting
9. Statistical Analysis
10. Snow Removal Schedule
11. Word Processing

### Comparative Considerations

In the use and allocation of any educational resource, it is necessary to make some tradeoff decisions concerning alternatives. In the case of the use of computer-based management applications, some of the considerations favor the use of manual administrative systems, others favor large systems, and still other factors favor microcomputer applications.

The following is a list of some of the advantages of microcomputer-based management applications:

- Equipment Cost. A microcomputer with 48K, dual disk drives, card reader, and a printer can be purchased for less than $5000.
- Ease of implementation. A school district can get started with a microcomputer with very little investment of time, effort, or money.
- Ease of operation. Microcomputers are relatively easy to operate; as such, highly trained technicians are not required.
- Flexibility. The school user can achieve the capability of downloading or extracting portions of a district's data base from a large computer to a microcomputer.
- Multi-purpose use. The same microcomputer can be used for administrative and instructional applications. By having multiple units within a district, there would be backup in the event of equipment problems.
- Operational responsibility. District staff are responsible for hardware operation and maintenance; for securing or supplying hardware; and software selection and maintenance.
- Software cost. The cost of purchasing commercially developed programs for microcomputers is much less than programs for larger computers. Programs for microcomputers can be sold in much larger volume than can programs for larger computers, and they are typically less complex.
- User control. The local school district can own the equipment, use it exclusively for their use, and completely control its use and operation.

The disadvantages of using the microcomputer for school management applications are:

- Available software. Most current administrative programs have been developed for large computers. There are very few administrative programs with documentation currently available for microcomputers.
- Difficulty of application development. It is relatively easy to develop fairly simple programs for microcomputers; however, more complex programs, in particular those that make extensive use of files and require extensive data manipulation and updating, are difficult to develop on a microcomputer.
- External reporting. As compared to larger computers, external reporting and integration are more difficult with stand alone microcomputers.
- Integrated applications. Due to the limited size of core and mass storage, it is not feasible to develop integrated data base systems by using microcomputers.
- Limited usability. Some management applications require large machines to run (computer scheduling of students, large sorts, large volumes of data storage) and are not amendable to small computers.
- Reliability. There are some problems with reliability of data storage on diskettes commonly used with microcomputers.

### Development and Support

Traditionally, the development of computer applications and support services to end users have been provided by computer services staff who are part of the same organization. They typically have been staff at the local school district level for those school districts large enough to support their own mainframe computer and its staff. Recently, some states and/or regions have developed software for use on a large computer. The operations and user staff have been affiliated with another level. For example, in Minnesota administrative software has been developed at the state level by MECC; however, the operation of the services centers and the end user support have been provided at the regional level. As microcomputers become more capable, it is obvious that the actual operation will be at the local school level. The question remains as to who will develop and support the software.

It is likely that software will be developed by a number of entities. Local school personnel will develop some of their own applications programs. However, this can prove to be quite costly since the time it takes to design, program, and test comprehensive systems (e.g., financial accounting) is not proportional to the hardware on which it is operated. Regional and state systems development teams will also be involved with the design and development of microcomputer-based management applications. Software firms, computer vendors, and publishers of electronic materials are also likely to get into the act. Because this type of software will be high volume, low unit cost in nature, it will typically take a relatively large distribution base to profitably market it. As these software products are "packaged," they necessarily must be generalized in order that a wide variety of users will be able to utilize them. Users will have to forfeit some of their flexibility in order to use these packages. With the economies of the situation, it may be a matter of obtaining 90 per cent of a desirable system versus not having computerized management systems.

It is likely that users of microcomputer-based management applications will receive support services in the form of training, trouble shooting, and other user services from a variety of sources. In addition to local district services staff, there will be regional, and perhaps even state, level support staff. Those new participants in the software development business - the vendors and publishers - will provide training workshops and seminars, hot line services, and very thorough user manuals. However, it is likely that end users will have to be more self-reliant than they have been in the past. However, the advantages and the economies of use will see widespread use of microcomputer-based applications for school management purposes in the not too distant future.

# MICROCOMPUTERS IN EDUCATION

Title IV-c Project
Bill Heck, Project Director
Robbinsdale Area Schools
Independent School District 281
4148 Winnetka Avenue North
Minneapolis, Minnesota 55427

Microcomputers in Education is a three-year development project, funded by a Title IV-c grant. The main focus of the project is to develop and classroom-test microcomputer programs for the elementary school curriculum. Programs have been developed for math, language arts, social studies, science, music, and foreign languages. The second and third year of the project includes short-term and long-term research on the effectiveness of the programs.
(All programs are available for the Commodore 8K PET computer. Some programs are available for use with the APPLE computer.)

## PROGRAM DEVELOPMENT

Authoring, writing, coding, and classroom testing summarize the general structure that is used to develop a computer program within the project.

### Authoring:

Ideas for programs will be primarily generated by classroom teachers or subject matter specialists along with the project staff. Many ideas for programs have come out of in-service courses available to teachers within the Robbinsdale schools.

Each new idea is then developed by the project director and facilitator. Many items need to be defined. Such as:

1. What content will be included
2. What things can be done with the content to take advantage of the power of the computer
3. What kind of presentation and screen format is required for the level of the student for which the program is intended
4. Are graphics appropriate
5. What kind of positive reinforcement should be used
6. What kind of help should be provided when the student gives an incorrect answer

### Writing:

This phase of program development is involved in translating the information from the previous section into a flow chart with a logical flow. This flow chart may be done by the project facilitator or by one of the student programmers.

### Coding:

This part of the development process is generally done by student programmers with assistance from the project facilitator. Special attention is paid to the use of standard subroutines for such things as input statements, title pages, stop routines. By allocating some of these items to standard subroutines, a third of a new program may already be completed.

### Classroom Testing:

Each program goes through several phases of testing. The first is in the project office. Each program is examined by the project staff throughout the coding phase. Upon completion, a copy is given a final run through by the project secretary. The program is then sent out to three key teachers to use with students. This phase takes two to three weeks. The key teachers document good features as well as any problems which may have arisen.

When all of the information is returned to the project offices, an evaluation is made by the project director. Changes may need to be made on the program or it may be ready for distribution to the 17 schools within the district. After a two- to three-month period if no problems arise with the program, it is made available to other school districts.

## IMPLEMENTATION

Each of the 17 schools involved in the project has a key teacher. This teacher, along with the media generalist in each building, are the contact people for the project.

All of the computers along with the programs are checked out of the media centers. This allows the use of an already established checkout procedure. Each building has a notebook which contains an alphabetical listing of all programs, as well as a subject listing of all programs. The third part of the notebook contains a documentation sheet on each program. This sheet gives the important information and directions about the program, as well as a sample run exactly as it appears on the computer. The value of a complete documentation sheet is that the teacher does not have to run every program in order to decide whether or not to use it.

In-service has been provided for staff in two ways:

| | |
|---|---|
| Total Staff: | Each elementary staff person at the 17 sites received two half-hour sessions on the project. Each key teacher participated in two half-day workshops. |
| In-Service Courses: | Ten 16-hour in-service classes on the computer were held. Approximately 320 staff people took these classes. |

## INSTRUCTIONAL CONCEPTS

Along with the exploration of computer uses in all elementary school subjects, various kinds of instructional concepts have been examined. A description of these concepts follows:

### Kid Proofing Programs

A basic philosophy of the project is that the computer should be used in the classroom by teachers and kids. Most of the programs developed within the project contain features which makes it impossible to get a computer systems error message. Through some subroutines within the programs the stop key on the PET computer is disabled and the input characters are controlled. This insures the teacher that students cannot accidently exit the program or distrub the screen display.

### Personalization

Initially giving the computer human characteristics seemed desirable. After several months of usage it now seems more desirable to view it as a machine. Statements such as "I like you" and "thanks for working with me" have all been deleted from the programs. This type of personalization takes up space and offers nothing instructionally. At first it also seemed desirable to have the computer ask "What is your name?" and to use the name throughout the program. However, after using several programs with this feature the students tended not to respond with their name but other undesirable words. Our present programs make reference to students by "Player 1" or some other technique controlled within the program.

### Positive Reinforcement

Positive reinforcement can be done in many ways, from using simple words to showing something spectacular on the screen. Inasmuch as possible some reward should be given. Too frequently the reinforcement for a correct answer seems to be the next problem. It is also necessary to minimize what may occur in response to an incorrect answer. If something quite spectacular happens when an incorrect answer is given, students will purposely get wrong answers. The programs from this project do not reinforce incorrect answers in any way.

### Program Duration

Several concepts were examined and used. Some programs never end. The computer randomly selects items for use and continues to do so until the computer is shut off. Teachers like this concept particularly at the kindergarten through third grade levels. Students work at the computer for a specified time limit. Some of the programs are set up to give a specified number of items, usually ten. Other programs allow the teacher or student to select the number of items. A summary is usually given at the completion of these programs.

### Whole Class Involvement

Although most computers and computer programs are meant to be used by one or two individuals, there are times when working with an entire class is desirable. A computer at each school site has been equipped with an output jack so that one or more large video monitors can be used. A program can then be introduced to a whole class or used with the whole class as an instructional activity.

### Programs Requiring One, Two, Three, and/or Four Students

Most programs seem to be written for a single student. Two or more students may use such a program by taking turns. An additional concept is to write programs requiring two, three, or four students. Positive feedback has been received from teachers regarding these programs requiring more than one student.

### Game Aspects

Students seem to prefer some kind of a game context. Two types have been used within the project. One allows students to participate in a game after giving a correct answer. The second is to put the instruction or practice within a game format. The second type of game context is preferred.

### Competition Between Students

A game situation involving competition between students is quite popular. Techniques have been explored which would allow two students with different abilities to use the same program. Programs involving competition can be used with small groups or the whole class for discussions and decision-making.

### Cooperation Between Students

The concept of cooperation was built into a program in the following way. Two students are given a time limit to alternately give correct answers to math facts. Each time a correct answer is given a part of a rocket ship appears. If they can complete the rocket ship within the alloted time, it will fly. If not, the ship dissappears and they must start over. More must be done with this concept.

### Graphics and Screen Motion

Programs can be done with or without using the graphics capability of the computer. However, using pictures as a part of the instruction as well as for reinforcement has a high priority within the project. The use of motion on the screen can be used to illustrate instructional concepts. The use of graphics and screen motion needs to be exploited even though the programming is more difficult and takes considerably more time.

### Sound

The use of sound as part of instruction can be very effective, particularly for music. Seeing the notes on the screen and hearing the appropriate sound has been received very favorably by music teachers. Sound has also been explored for positive reinforcement. However, such reinforcement can become monotonous as well as disturbing if a whole class must listen to it.

## RESEARCH

Five research studies are being conducted as a part of the project. Information on each study is as follows:

**Study One:** Introduction to Fractions
**Grade Level:** First
**Time Period:** 2 weeks -- March 23 through April 3
**Classrooms:** Five
**Computer Programs:** Fraction Concept, Fractions 1, Fraction Symbols 1, Fraction Symbols 2

**Study Two:** Multiplication Facts
**Grade Level:** Third
**Time Period:** 3 weeks -- March 2 through March 24
**Classrooms:** Four
**Computer Programs:** Math Drill, Mult. Facts 0-25, Zap Mult. 2-6, Zap Mult. 5-9, Math Swim, Baseball, Dragstrip Math, Mult. Facts 0-100, Mult. Table, Tic-Tac-Facts

**Study Three:** Nouns and Verbs
**Grade Level:** Fifth and Sixth
**Time Period:** 4 weeks -- March 9 through April 10
**Classrooms:** Five
**Computer Programs:** Nouns-R, Verbs-Action, Verbs-Helper, Verbs-Being, Verbs-Review, N-V Pattern, N-V-N Pattern, N-LV-N Pattern, N-LV-Adj. Pattern, N-V Pattern Review

**Study Four:** Fractions (concepts-equivalent-adding)
**Grade Level:** Sixth
**Time Period:** 3 weeks -- March 9 through April 10
**Classrooms:** Four
**Computer Programs:** Fractions 2, Compare Fractions, Fractions-Equiv 1, Fractions-Equiv 2, Common Denominator, Reducing Fractions, Add Fractions 1, Add Fractions 2

**Study Five:** Cherry Creek Spelling
**Grade Level:** Fourth thru Sixth
**Time Period:** 4 weeks -- March 9 through April 10
**Classrooms:** Five
**Computer Programs:** Cherry Creek 10 Program Set:
Cherry Creek A0-E9
Cherry Creek F0-H9
Cherry Creek I0-J9
Cherry Creek K0-L7
Cherry Creek L8-M9
Cherry Creek N0-N9
Cherry Creek O0-O9
Cherry Creek P0-P9
Cherry Creek Q0-Q9
Cherry Creek R0-R9

No data is available on the studies since they are being conducted during the writing of this paper.

# A CLEARINGHOUSE FOR EDUCATIONAL APPLICATIONS OF MICROCOMPUTERS

Donald C. Holznagel
Coordinator, MicroSIFT
Northwest Regional Educational Laboratory
710 S.W. Second Avenue
Portland, Oregon 97204

MicroSIFT, a clearinghouse for microcomputer-based applications in education, is established at the Northwest Regional Educational Laboratory. It is engaged in collecting, organizing and disseminating information about such applications. It has designed and field-tested an evaluation model for instructional applications based on the cooperation of a network of centers of educational computing.

## INTRODUCTION

Microcomputers are being acquired in increasing numbers by schools. Over the last decade there has been a high level of funding and development to support the great potential computers have repeatedly demonstrated in the instructional environment. In spite of these factors, a majority of schools have not taken advantage of the many possible applications of computers.

A 1979 survey of schools in the northwest states, done by the Northwest Regional Educational Laboratory (NWREL), identified a crucial obstacle to wider use: lack of adequate materials. A lack of documentation, validation and dissemination have been cited as factors contributing to the apparent scarcity. This lack of adequate materials is found in both administrative and instructional computing applications.

Other often cited hindrances to significant use of computers in education are: a lack of skilled personnel to provide materials and training; lack of information concerning the use of computers in curricular areas; technical problems related to hardware/software incompatibility; and cost.

A clearinghouse for microcomputer instructional software and teacher information has been established at NWREL under a contract with the National Institute of Education. The clearinghouse, MicroSIFT (Microcomputer Software and Information for Teachers), is a central project of NWREL's Computer Technology Program. The Program has been productively involved in developing methods and materials for the use of computers in education for more than a decade.

The key objectives of the MicroSIFT clearinghouse are:

1. To develop and implement a model for the dissemination of microcomputer software, information, and materials for educational use at K-12 levels.
2. To develop, validate, and implement an evaluation model suitable for computer-based instructional packages. The model will include support of the long-term assessment of the quality of materials proposed for distribution.
3. To develop and implement a user support and technical assistance program in the Northwest region, which will increase the capability of local and state education agencies to choose and implement microcomputer applications in education.
4. To develop and implement a feedforward model for guiding and directing development of new computer-based instructional materials.

The MicroSIFT clearinghouse procedures and services have been designed by Dr. Judith Edwards, Director of the Computer Technology Program and Donald Holznagel, Coordinator of MicroSIFT, with significant input from the staff of CONDUIT, an Iowa-based clearinghouse for software in higher education, two MicroSIFT advisory boards (a National Technical Advisory Board and a Regional Users Advisory Board), and other consultants.

The scope of the project includes both

instructional and administrative applications for education. The initial focus of staff efforts has been on instructional applications because of the high interest and available packages in that area. The project has also been collecting information on non-instructional software, and will be examining possible approaches to evaluating those materials during 1981.

## ORGANIZATIONAL BASE

Established educational institutions and consortia will be the major initial mechanism for the collection, evaluation and dissemination activities of MicroSIFT. A network of organizations is being established consisting of existing computer centers which serve a large number of schools and provide significant work in instructional applications of computers. Their personnel are experienced in computer-based education and in dealing with the technical problems involved in program transfer. Such centers have a need to evaluate the efficacy of instructional software. Many have done evaluations on a small scale, but are not able to support large-scale continuous evaluation efforts. They will serve as long-term sources of materials and information. In addition, these centers will serve as agents for review, product documentation and finishing, field test sites, and further development.

The Network is being established in two stages. In the first stage, a selected set of 15 centers was enlisted as initial members. They were chosen because they had several years of involvement in instructional computing, including development and evaluation of computer-based materials, provision of inservice for teachers, support of microcomputer implementation in schools, and having full-time instructional computing staff members. An initial planning meeting for these organizations was held in October, 1980, at the Jefferson County Schools office in Lakewood, Colorado. At that time, a memorandum of agreement was drawn up, and potential activities were discussed.

The second stage of the Network will be based on successful activities in the first stage. Alternatives for the second stage will be discussed at a meeting of Network participants in May, 1981. A major expectation is that criteria for an expanded membership will be determined then.

The MicroSIFT Network will be the major vehicle for the evaluation process and the source of reviewers and test sites. The centers offer access to experienced and beginning users, qualified summary reviewers, a variety of student groups, geographic locations, and ethnic, minority and socio-economic groups. They are experienced in evaluation of computer-based materials, and almost any desired hardware base can be found in use among them. During January and February, 1981, the initial Network members conducted the field test of the courseware evaluation process and instruments described in the next section of this paper.

## EVALUATION MODEL

It is encouraging to note that many individuals, small and large publishing companies, and hardware manufacturers are beginning to produce programs and materials for microcomputer applications. However, a primary need of the educational consumer surveying those products is a basis for assessing educational value and utility. The basis for good evaluation of materials is the existence of a well-defined process supported by usable instruments validated for the environment of their intended use. A primary concern of MicroSIFT to date, in line with the second objective, has been the development of such a process and instruments uniquely suited to microcomputer-based instructional materials. MicroSIFT staff were joined in this effort by the Program for Research on Evaluation of NWREL, which is exploring new evaluation techniques and environments.

Sample instruments (checklists, questionnaires, rating scales) used by schools, regional service centers, microcomputer publications and individuals were collected and analyzed. The CONDUIT evaluation procedure and instruments weighed heavily in the initial design. Discussions with organizations having significant experience in the instructional application of computers in the K-12 environment also contributed substantially to the development of this evaluation model.

The evaluation procedure presently consists of four stages:

1. Sifting. An initial judgment is made whether a program or package is suitable for the K-12 microcomputer environment and is ready for evaluation. Certain fundamental criteria must be met: e.g. does it have instructional value (as opposed to strictly recreation)? Can it be run on a microcomputer with little or no adaptation? This step is to be completed by MicroSIFT staff.
2. Description. Factual information about the program and materials is reviewed: e.g., instructional purpose, technique, package form, documentation available. This information is collected and recorded by MicroSIFT staff and by the reviewers.
3. Evaluation. Two or more professional teachers at a Network site with experience in the subject area and intended grade level of the package review the material, test the program, and make judgments concerning its value as an instructional package. A summary review is completed by a person from the Network center with experience in development and evaluation of computer-based instructional materials.
4. In-Depth Evaluation. When a package successfully passes the first three stages and is rated high enough to warrant further evaluation, instructional effectiveness will be assessed through pre-testing and post-testing, student observation, or a combination of these methods. Some instruments and procedures for observation and analyzis of student-computer interaction are in design and development. This stage was not included in pilot or field tests.

The resulting evaluation model is in its fourth version. Version 1 was pilot-tested using 18 programs with a variety of non-commercial sources by a group of six teachers from the Portland Public Schools in Oregon. Three people who are full-time professionals in instructional computer applications were chosen from the Portland area to serve as summary reviewers. Version 2, based on that experience, was tested using over 50 commercial and non-commercial programs by a group of eight teachers in the Beaverton Public Schools in Oregon. Coincident with the development of Version 3 of the evaluation instruments, and using information from the pilot experiences, an Evaluator's Guide was developed to assist evaluators in interpreting the criteria statements in the evaluation form.

After professional review, a revised draft was produced and used by the Network sites in the field test of the evaluation process during January and February, 1981. Twenty microcomputer-based instructional packages were evaluated in the field test. In addition, the Evaluator's Guide was tested in a university course in Instructional Design at Utah State University, Logan, Utah.

## FEEDFORWARD MODEL

The three major components in the feed-forward model referenced in the fourth objective of this project are as follows:

1. Identification. Specific programs, packages, or instructional units as well as major curricular areas must be identified as definitely needed or as potentially fruitful for development activities.
2. Organization. The information concerning needs must be collected and organized into meaningful patterns, with specifications developed.
3. Dissemination. The specifications of needs must be disseminated to publishers and other developers in both the commercial and non-commercial sectors.

The process of identification of needed materials is to operate on both the general and specific levels. At the general level, areas of curriculum will be identified which have high potential for employing microcomputer materials successfully, and which are not currently well-served with such materials. Much of this information will come from the members of the MicroSIFT Network. Solicitations also will be made through the newsletter, and by direct letter to publishers. At the specific level, programs or packages will be specified to support student learning at a particular point in a course to meet a certain objective. This will involve the design of forms and the provision of processes for distribution and return. It is anticipated that in many cases, ideas for needed development will come from a teacher who is involved in microcomputer use.

MicroSIFT staff will provide a categorization scheme through which the information previously identified can be organized for dissemination. The three major dimensions will be Grade Level, Subject Area, and Mode of Use.

A publication entitled the Developer's Advisory will be produced which will include both general and specific development needs as identified by the processes above. The Advisory will be disseminated through the avenues described under DISSEMINATION, and by direct mail to a list of publishers and other developers now being assembled by MicroSIFT staff.

A Developer's Guide containing standards, procedures, and techniques is planned to be developed for dissemination to the same audiences to provide support and direction for improvement in the quality of microcomputer-based material.

## DISSEMINATION

The final version of the Evaluator's Guide and instruments resulting from the field test is being published by NWREL. Information about its cost and availability may be obtained from the Marketing Office, NWREL, 300 SW Sixth Ave., Portland, Oregon 97204. The evaluations from the field test are published in MicroSIFT News, March, 1981. Publication of reviews will continue on a quarterly basis, although permanent avenues for their publication have not been determined.

Information concerning clearinghouse plans, activities, progress, services, and outputs will be made available in several ways. One of the primary avenues is the expanding general dissemination capacity being directly supported by NIE: Regional Exchanges, Regional Service Programs and State Capacity-Building Programs. The CEDaR Comments and Educational R & D Report, publications of the Council for Educational Development and Research, will be used for spread of information to the readers of those publications. A quarterly newsletter, MicroSIFT News, has been established to facilitate spread of information through direct mail contact with a continually growing mailing list. News releases and articles will be prepared for publication in professional magazines which are specifically aimed at the educational computing community.

# LIFELONG LEARNING AND THE POST-INDUSTRIAL REVOLUTION

by
Paul Hurly
Continuing Education Division
The University of Manitoba

Virginia Davis
Instructional Media Services
Manitoba Department of Education

ABSTRACT: The application of high technology in society has been heralded as the coming of the 'post-industrial revolution'. Innovations in computer and telecommunications technology have spawned a new technological family - informatics. Informatics refers to computer communications of all kinds. The Canadian Government has developed a version of this technology called Telidon. The impact of systems like Telidon may be felt in many sectors of our society. In Manitoba a small group of educators ha begun to experiment with the application of Telidon and other informatic systems to create more open learning systems. The need for lifelong learning to meet the demands of a rapidly changing society has grown dramatically during the past decade. The application of systems like Telidon may provide a tremendous impetus to the promotion of lifelong learning in western society. This paper will review the rationale for lifelong learning, and will suggest the potential impact of informatics on continuous learning based upon the current work of the authors, and by examining developments on the international scene.

## Introduction

The concept of lifelong learning has become paramount in western industrial society during the past 10 years. It fosters an image of human beings and their growth which subsumes all progressive thinking concerning the nature of education. Championed by such organizations as Unesco and the Organization for Economic Cooperation and Development (OECD), lifelong learning has reached its fullest expression through the vision of a "learning society".

The challenge and the need for lifelong learning has never been greater than it is today. High technology has begun to unleash the second "great transformation" which promises to radically rewrite the course of history and the form of western social structures and systems. The microprocessor, the videodisc and advanced telecommunications networks have imparted an even greater velocity to social change. The post-industrial revolution is upon us. It has rendered already some knowledge and skills obsolete, and it has given us cause to question some of the basic precepts of institutional schooling.

This paper will provide an overview of lifelong learning as an educational concept, and will discuss some of the ways that videotex technology may facilitate lifelong learning during the next two decades. To accomplish this we will be drawing partially on our experience with the application of Telidon, a viewdata system, to educational programs in Manitoba. We will briefly report on our work and we will summarize information concerning some of the other international videotex field trials. While tracing some of the roles which videotex might play in education we will also identify several strategic issues which promise to influence the adoption of videotex by society.

## The Second Great Transformation

There is little doubt that western society is being deeply affected by the expanding application of computers. From its initial development as an information processing machine to its present penetration of manufacturing for automated production; medicine for micro-circuity prosthetics; and, school as adjunct instructional media, computers have begun to change our lives.

Karl Polanyi's seminal publication, *The Great Transformation* (1944), has traced the phenomenal alteration of 19th century European society by the machine. Europe industrialized with the application of incredible power tools and was swept at a dizzying speed, Polanyi recounted, into a period of social and economic change based upon the economic principle of self-interest. The result was a catastrophic dislocation of the lives of the common people, social upheaval, and attempted rebellions throughout most European nations during a 30-year period.

Sociologist Daniel Bell has coined the phrase "the post-industrial society" (1973) to identify the period the western world has now entered. This era Bell characterized by a shift from goods-producing to a service society; centralized application of theory for innovation in technology; the widespread application of computer data processing and automation in the economy; and, the creation of a new intellectual science to design the systems for the new technology.

The British report *The Manpower Implications of Micro-Electronic Technology* (1980) envisaged an evolutionary as opposed to a revolutionary transformation of society due to the application of high technology. Yet it acknowledged that the job-loss impact will be felt hardest by that sector already hurting in the economy - the unskilled worker - as well as the service sector. Like many industrialists and other international studies (Canada, 1979; Science Council, 1980; Sweden, 1979) the report conceded that automation is essential for the present national economy to remain competitive in the future.

Other authors, such as Alvin Toffler (1980) and James Martin (1978), have provided a scenario analogous to Polanyi's assertion that the impact of new modes of production on society more closely resembles the metamorphosis of a caterpillar. Marc Porat's study (1977) of the US economy has shown the phenomenal growth of the American information sector, from 24.5 percent of the labour force in 1930 to 46.6 in 1980. This sector in 1967 accounted for 46 percent of the American GNP. Robin Crickman (1979), in her study of the "emerging information specialist", predicted that more than 50 percent of the US GNP would be absorbed in information management by 1980.

While some forecasters scoff at Toffler's scenarios other useful analysis (Pratt, 1980, pp. 62-7) has shown that not only are many predictions about the impact of computers on society proving to be true, they are coming to pass much earlier than predicted. This impact will be felt, and must be responded to, by society's educational systems.

## Learning Without End

The concept of lifelong learning is not new to human society. Traditional societies recognized that the entire community was involved in the process of preparing each new member for entry into the various roles and responsibilities which the society expected its adults to assume. Moreover, each adult was clearly acknowledged to pass through levels within the society, each level preparing the individual for the next.

Today, the lifelong learning concept, as embodied in the image of the "learning society" championed by adult educators, maintains that learning which begins in school must continue throughout adulthood. Learning in such a society will occur through and be encouraged by all social organizations, and the economic system will place as high a value on learning as it does on productivity (Lowe, 1975; Faure, 1972).

Martin argued that adults in the future will have to prepare for three or four careers in a lifetime. The important question for educators has become 'education for what' as opposed to 'how best to educate'. In France, Jacques Hebenstreit, a department head at the Ecole Superieure d'Electricite, has reported that the purpose of education has been redefined to emphasize the need to help people to adapt to a future we cannot predict today. The imparting of learning, adaptation and other intellectual survival skills thus becomes crucial. Knowledge for its own sake is downgraded.

David Pratt has stated that today "we are educating adequately for the 1960s". The children in second grade in 1981, he has argued, are not being prepared so they can cope with the future. We cannot begin to respond today to the infinite details of a life ahead which we cannot fully imagine. Schools, Pratt believes, must be more effective in preparing students for adapting to the future. The Council for Educational Technology in England has identified several purposes for education to help people develop survival skills. People need a survival kit to help them to adapt to changing life roles, to increased leisure time, to evolving social patterns, to a changing self-concept, to the quirks of their physiology, and to acquire the requisite civic and literacy skills for their society (Large, 1980).

In a more extreme position Ivan Illich has argued (1973) that education must no longer be viewed as a commodity. Learners must be given access to "convivial tools for self-initiated learning" (p. 69). Society must be de-schooled (1971). Illich noted that "professional teachers laugh at the idea that people would learn more from random access to learning resources than they can be taught" (p. 70).

The work of researchers like Allen Tough (1979) is changing educators' notions of how, when, where and why adults learn. Tough and co-researchers found that the average adult undertook eight learning projects per year totalling up to 800 hours. Less than one percent of the projects were for formal credit. They also identified various reasons why such learning projects experience difficulties. One major recommendation was that more support services are required to help learners to plan their learning and to direct them to appropriate resources. Yet in Canada, as in many western nations, it is the institutional sector controlled by professionals which garners most of the educational dollar. Adult participation in institutional continuing education in Canada increased 67.7 percent from 1961-62 to 1971-72 (OECD, 1976). Yet Tough maintained that roughly 70 percent of all adult learning is self-directed.

"The pace and direction of future change in society will be influenced," Tough argued, "by the adult's resistance to change. Changes in society will, in turn, result in people learning certain knowledge and skills that are not common at present" (p. 40). This was certainly the case as Polanyi pointed out during the 19th century. The Industrial Revolution also dramatically shifted responsibility for education from the household to schools. The post-industrial revolution will not only require new knowledge and skills to be mastered, but promises as well to affect the location of work and education, as well as further to increase the amount of leisure time (Toffler, 1980).

We have begun to witness a gradual change of attitude concerning the education of adults. Following the Canadian government's report on Paid Educational Leave (1979) some 200 collective agreements in Canada now contain PEL clauses. More and more Canadian industries, businesses and government departments are providing educational benefits for their employees. These services range from job-related courses, to personal development and liberal studies programs. Open access is gradually being promoted to higher and technical education via multi-media delivery, mature age entry and flexible scheduling. The challenge will be to meet the growing need for lifelong learning with more creative and flexible means. In Manitoba we have begun to experiment with a videotex computer-based communication system to enhance both formal in-school and lifelong learning.

## Videotex Services

Since the early 70s a new form of telecommunications technology has been evolving in Europe and North America. Standards have been established for this new industry only recently so the terminology is often ambiguous and confusing. By the term *informatics* we are referring to computer communications of all kinds. One specific term, *videotex*, has been used in some quarters as a synonym for informatics, but more generally it has come to be used to identify a type of computer communications which can utilize the versatile home TV cathode ray tube as a display screen, thus creating a mass market. There are two basic variations of videotex technology.

*Teletext* can be obtained as both a non- and pseudo-interactive videotex system, while *viewdata* (sometimes also termed videotext, just to really confuse) is an interactive system. Teletext is broadcast through the air like a regular TV signal. Viewdata operates only in a transmission mode over paired plant wire (voice grade telephone line), coaxial cable and fibre optic lines. Information for both systems is stored in a computer.

An example of non-interactive teletext would be the news and weather bulletins superimposed on your TV screen during the broadcast of a regular TV program, or the recent introduction of captioned news by the ABC network. In a pseudo-interactive teletext system the user at home or in the office has the potential to access up to about 400 screens or "pages" of data which are broadcast in a continuous, repetitive cycle. The user command instructs the decoder on the modified TV set to pluck the desired page from the teletext broadcast cycle. The teletext user must therefore wait for the data requested to be broadcast in the cycle before the home TV can display the information. At no point does the teletext user communicate directly with the computer storing the data.

By contrast, a viewdata system user can access immediately data anywhere in the entire main computer's memory, which could contain up to 500,000 pages of data, or be routed to other specialized data bases, via an on-line network analogous to modern office or scientific data retrieval systems. The videotex user or viewer selects what is to be displayed by pressing a hand calculator-like key pad or a key-board device to input the specific page code desired, or simply to turn the pages in sequence (Cherry, 1980; Sigel, 1980).

The first domestic application of teletext technology was conducted by the BBC in England with the introduction of CEEFAX in 1976. ORACLE is another version of teletext. CBS recently sent shock waves through the international telecommunications community when it chose to utilize the French ANTIOPE teletext system for a field trial which started in Los Angeles in January 1981 (BM/E, 1981).

The first commercial version of viewdata was also introduced by the British. The British Post Office Prestel system was heralded as the standard for the industry when it was unveiled in 1979 (Sigel, 1980; Urrows, 1981). The French ANTIOPE-Videotext provided some competition but it was Prestel which won a lucrative field trial in West Germany which started in 1980. France has now released a new viewdata system, Teletel, and the Japanese are testing two versions at this time which could prove to be highly competitive on the international market. The introduction of Telidon by Canada in 1979 caused a major stir among its European competitors for the Canadian viewdata system offered superior graphic resolution, greater colour choice and separation, and foreward and backward compatibility with various terminal systems - properties which the European versions couldn't match. As a result Canada won major business in Switzerland and Venezuela, and is undertaking field trials in Washington, D.C. and in New York, Oregon and Washington states.

In the United States Warner Telecommunications introduced a field trial of its Qube teletext system in Columbus, Ohio in 1977. Knight-Ridder and AT&T have developed the Viewtron viewdata system. Their subsidiary, the Viewdata Corporation of America, has a field trial underway in Coral Gables, Florida and will announce shortly plans for a national initiative. GTE Telenet Infovision represents the first plans for a national videotex service in the US. Both the Tandy Corporation and Apple are exploring means for using their microcomputers as smart terminals for a videotex service, while Zenith and Microtel, to mention only two companies, have terminals available which will offer versatile reception of a variety of broadcast or transmission videotex and TV signals.

The Manitoba Telephone System has undertaken a series of network design trials in conjunction with the federal Department of Communications. These trials, which started with Project Ida in Headingley, a small town west of Winnipeg, are exploring the use of coaxial cable and fibre optics for transmission lines, the use of microprocessors as network switching devices, and the application of home information system technologies (Hlynka and Hurly, 1981). Telidon is one of the services offered to home owners in the field trial.

The Winnipeg St. James School Division, in cooperation with MTS, has placed several types of teletext and viewdata terminals into selected schools and is accessing several different data bases in order to investigate the application of an on-line system for the delivery of Computer Assisted Instruction (CAI) services as part of the regular school program (Groff, 1980). The Manitoba

Department of Education has been an active observer of this experiment. The University of Manitoba Continuing Education Division has created more than 100 pages of data ranging from campus news and activities, to an independent study course and public service information.

## Impact of Videotex on Learning

Marshall McLuhan (1964) and countless other media and social critics have identified the major impact which TV has had on how people learn, what they learn, how people use their leisure time, and the concept of community. It is possible that the concept of "television as meeting-place" (Gregoire, 1979) and as a social recreation and entertainment facility (Perspectives Canada, 1980) will be reinforced by the mass penetration of videotex services during the next two decades. Videotex services have epitomized for many people the image of the "electronic cottage" created by Toffler. Whatever its final form, videotex represents the potential convivial tool which could help to decentralize and demassify society, and which could support - if one of Toffler's futuristic scenarios is accurate - a means of permitting more work and play to occur in the home. In a service oriented, post industrial society videotex systems could provide a new and essential means to assist the population to adapt to change.

Videotex systems could provide various forms of CAI programming (drill-and-practice and programmed instruction); computer managed instruction; experiential or discovery learning; programming and algorithmic formulation; problem solving; simulations and modelling; visualization by accessing computer, motion, still or microfiche media; and information retrieval and processing (Aiken and Braun, 1980). The Integrated Home Information System concept could allow learning to occur in a variety of forms. Using videotex on-line or videodisc provided programs learners enrolled in formal courses could progress through their lessons employing distance or independent study models. Education could occur in the home, office - anywhere a terminal was available (Hiltz and Turoff, 1978). Most educators agree, however, that CAI will remain an adjunct to correspondence or teacher-based instruction.

Telidon will offer by 1982 several features which excite distance education course designers. An electronic mail service would allow learners to send their assignments rapidly to the instructor for grading. Terminal-to-terminal communication would add an important interactive component to instructional tutoring in distance education. An audio-graphic capability will also facilitate simultaneous voice and image instruction (Hurly, 1981). Education also fulfills a social function, and research has shown that optimal learning occurs when two to three people can interact regarding CAI material. Thus the promotion of group learning in homes and community resource centres equipped with terminals and other hardware could be an important future continuing education direction. Confederation College in Ontario has been experimenting with microcomputer and videoconferencing networks to reach groups of learners in remote centres in a cost efficient, learning effective continuing education program (Frewin, 1980). Similar programs have been utilized in California for several years.

The library service has long identified the utility of high technology for its administrative systems. The library of tomorrow can facilitate lifelong learning activities by developing open learning systems based on computer technology (Davis, 1980). Libraries could loan software and courseware to the public, perhaps even terminals and other hardware, and could expand their reference services and place these on a public data base.

Keeping in mind the need to promote self-directed study a videotex system could be used as a "non-human planner" (Tough, 1979) to help people to initiate their own learning programs. The CHOICES program provided by the Canada Employment and Immigration Commission via CSG Limited of Calgary, Alberta offers the person seeking career counselling advice access to a vaste computerized library of employment information. If such a service were available on a videotex network it would potentially be open to thousands of households. Vaste amounts of how-to-do-it and other information, plus lists of resource people, on a videotex data bank could open up a whole new realm of employment opportunity and learning activity for millions of learners of all ages.

## Potholes and Pitfalls

Anthony Oettinger and Nikki Zapol write: "How can technology best help people to learn as their own goals change along with social and economic conditions? The question is worth asking only if it is granted that technology can serve learning at all. ...Far from producing 'visible results within a limited period of time,' educational technology ... has made no significant difference in learning achieved through formal schooling" (1972, p. 7). No-significant findings, they claim, confirm the limitations of research on formal schooling and suggest that issues of technology are independent of learning methods and content.

The question of *control* over the means of learning, whether it be in schools, or in the broader forum of the community, thus assumes great importance. The assertion that technology does not affect learning flies in the face of common sense when one considers the impact of print technology. The question of control is crucial as we contemplate how videotex might enhance the future of learning. Illich noted that the alphabet and the printing press are ideal convivial tools. And yet the book has been gradually dominated over time by large, centralized market forces and put to use in education as a static one-way storage system.

Much of the formal education - schooling - which occurs today performs a transmission or mediation function. The process assumes there is a direct cause and effect between teaching and learning. The use of computers in education today is compatible with this notion of learning, as well as with the impersonal, autocratic nature of educational institutions (Ackoff, 1974).

Skinnerian psychology is implemented via CAI drills which at best provide limited ranges of choise. "Individualization" has been cited by countless educational critics as the primary means of reforming schooling in order to make formal education more humane and effective. Yet in the face of the many policy issues computers may be no closer to individualizing learning then any previous technology.

Unit costs decline as the "scale of aggregation" of an educational product goes up (Oettinger, 1969; Oettinger and Zapol, 1972). Thus policy makers must balance preference for variety to meet the needs of diverse groups of individuals versus the cost savings inherent in mass produced education. If lifelong learning is to benefit from videotex technology we must grapple with the problem of making "the transition from perceived needs unmatched by firm demand to an effective market capable of supporting diversity on a continuing basis" (Oettinger and Zapol, p. 18).

Related to the above discussion educators should also be aware of the recent controversy caused by Joel Rakow's article "The Audiovisual Myth in America" (1980). Rakow identified four myths implicit in the writing of Thomas Edison and other early media proponents, and he traced the operation of these fallacies through 60 years of writing and research on the topic of instructional media. Not only did he fail to locate any empirical data to transform the myths into facts, Rakow identified numerous instances of modern fabrication based upon the myths, or sheer hearsay. In light of Rakow's compelling analysis it is imperative that media specialists begin to curtail such practices as bandwagon demands for new hardware - like computers - and spreading unsubstantiated claims, or claims based upon unreliable research regarding the educational effectiveness of instructional technologies.

Another pitfall has been identified by the brilliant MIT computer scientist, Joseph Weizenbaum. Corporate advertising, he observed, has assumed the power to define the problem as well as the solution (1979). Thus people are informed they have body odor for which they need deodorant. A new product is launched! Both the current marketing of home microcomputers and the pending introduction of commercial videotex services might be viewed as analogous exercises in the corporate priming of consumption.

Communications media affect and are affected by the social environment. The ability of videotex to facilitiate learning which could promote human adaptation will depend upon the magnitude of the social upheaval during the next two decades. If the unskilled and service sector work force will suffer the major brunt of high technology related job displacement, then videotex educational services, to be effective in redressing this impact, must be accessible to the affected people and capable of meeting their immediate needs. This will be contingent upon the cost of the service, its distribution and the relationship between job training programs and the available areas of new employment. David Godfrey (1980) has suggested the diffusion of videotex services will be enhanced due to the use of the already omnipresent home TV as a receiver set. Conversely, the regulatory labyrinth in Canada and the US could significantly interfere with the spread and effective implementation of videotex technology on a national or continental basis (Grundfest and Baer, 1978).

Acceptance and utilization of videotex services will also depend upon the public perception of what danger on-line systems may pose to individual privacy and confidentiality (Salton, 1980), as well as the user public possessing the requisite skills and psychological predisposition to interact with a computer-based information processing system.

## Summary

Human society faces a prolonged period of major social and economic transformation brought about by the application of computer technology. Lifelong learning has been reaffirmed as an essential means of helping the total population to adapt to a rapidly changing world. Access to continuing and adult education programs could be facilitated by the introduction of a public videotex network, such as the Telidon system being tested in Manitoba, if the necessary hardware and software for sound CAI programming were made available. Various factors such as who controls the content creation, national regulation, economies of scale, diversity of the content, sophistication of the system, cost of the service, and perception of the technology's effect on individual privacy, will ultimately affect the market penetration and social utility of videotex.

## References

Ackoff, Russell L. Computer and education. In M. Marois (Ed.). Man and computer. New York: American Elsevier Publishing Co., 1974, pp. 127-36.

Aiken, Robert M. and Ludwig Braun. Into the 80s with microcomputer-based learning. Computer, July, 1980, Vol. 13, No. 7.

Bell, Daniel. The coming of the post-industrial society. New York: Basic Books, 1973.

BM/E. Testing teletext: CBS takes the plunge. Broadcast Management/Engineering, Jan. 1981, Vol. 17, No. 1, pp. 115-18.

Britain, Government of. The manpower implications of microelectronic technology. The Department of Employment Manpower Study Group on Microelectronics, London: Queen's Printer, 1980.

Canada, Government of. Telecommunications and Canada. Consultative Committee on the Implications of Telecommunications for Canadian Sovereignty, Ottawa: Supply and Services, 1979.

________. Education and Working Canadians. Report of the Commission of Inquiry on Educational Leave and Productivity, Ottawa: Labour Canada and Minister of Supply and Services, 1979.

Cherry, Susan. The new TV information system. American Libraries, Feb. 1980, pp. 94-110.

Crickman, Robin D. The emerging information professional. Library Trends, Fall 1979, Vol. 28, pp. 311-27.

Davis, Virginia. Telidon: Information teaser of the future? Manitoba Library Association Bulletin, Sept. 1980, Vol. 10, No. 4, pp. 3-10.

Faure, Edgar, et al. Learning to be. Paris: Unesco, 1972.

Frewin, Carlos. Outreach. Thunder Bay: Confederation College, 1980.

Godfrey, David and Douglas Parkhill (Eds.). Gutenberg two.(2nd edition) Toronto: Press Porcipic Ltd., 1980.

Gregoire, Reginald. Television and values in the educational project. Government of Quebec, Corseil Superieur de l'education, 1979.

Grundfest, Joseph and Walter Baer. Regulatory barriers to home information services. In Herbert S. Dordick (Ed.). Proceedings of the Sixth Annual Telecommunications Policy Research Conference. Lexington, Mass.: D.C. Heath and Comp., 1979, pp. 323-327.

Groff, Merrill. The little school on the 'Electronic Highway'. Contact, Feb. 1981, Vol. 7, No. 1, p. 4.

Hiltz, Starr R. and Murray Turoff. The network nation. Reading, Mass.: Addison-Wesley, 1978.

Hlynka, Denis and Paul Hurly. Manitoba site of futuristic telecommunications experiment. Videodisc/Teletext. Vol. 1, No. 2, 1981, (in press).

Hurly, Paul. The joy of informatics. Canadian Journal of University Continuing Education, Vol. VII, No. 2, Winter 1981, pp. 45-8.

Illich, Ivan. Tools for conviviality. New York: Harper and Row, 1973.

_________. Deschooling society. New York: Harper and Row, 1971.

Large, Peter. The micro revolution. Glasgow: Fontana Paperbacks, 1980.

Lowe John. The education of adults. Paris: Unesco Press, 1975.

Martin, James. The wired society. Englewood Cliffs, N. J.: Prentice-Hall, 1978.

McLuhan, Marshall. Understanding media. London: Rutledge and Kegan Paul, 1964.

OECD. Review of national policies for education: Canada. Paris: Organization for Economic Cooperation and Development, 1976.

Adler, H. J. and D. A. Bruregard. Perspectives Canada III. (for the) Office of the Senior Advisor on Integration, Statistics Canada, Ottawa: Minister of Supply and Services, 1980.

Oettinger, Anthony G. Run, computer, run: The mythology of educational innovation. Cambridge, Mass.: Harvard Un. Press, 1969.

Oettinger, Anthony and Nikki Zapol. Will technology help learning? Teachers College Record, Vol. 74, No. 1, 1972, pp. 5-52.

Polanyi, Karl. The great transformation. Boston: Beacon Press, 1944.

Porat, Marc U. The information economy: Definition and measurement. Washington: U.S. Government Printing Office, 1977.

Pratt, David. Curriculum design and development. New York: Harcourt, Brace and Javanovich, 1980.

Rakow, Joel. The audiovisual myth in America: Filling the cracks of educational technology. Journal of Education, Vol. 162, No. 8, 1980, pp. 34-54.

Salton, Gerard. A progress report on information privacy and data security. Journal of the American Society of Information Science, March 1980, pp. 75-83.

Sigel, Efrem (Ed.). Videotext: The coming revolution in home/office information retrieval. White Plains, N.Y.: Knowledge Industry Publications, 1980.

Sweden, Government of. New views: Computers and new media - anxiety and hopes. Report by the Commission on New Information Technology. Stockholm: 1979.

Toffler, Alvin. The third wave. New York: William Morrow and Comp., 1980.

Tough, Allen. The adult's learning projects. (2nd edition) Toronto: Ontario Institute for Studies in Education, 1979.

Urrows, Henry and Elizabeth. Prestel is working in Britain. Educational and Institutional TV, Jan. 1981, pp. 27-30.

Weizenbaum, Joseph. Once more: The computer revolution. In Michael L. Dertourzos and Joel Moss (Eds.). The computer age: A twenty-year view. Cambridge, Mass.: The MIT Press, 1979.

# REALITIES OF TRANSPORTABLE MICRO-COURSEWARE WITH GRAPHICS EMPHASIS[1]

Gerald L. Isaacs
Waukesha, Wisconsin
Carroll College

Graphics is being increasingly used in education and can be a very powerful tool. The graphics available on today's microcomputers can be a valuable aid to educators. However, good graphical courseware is not widely available. Courseware is being written and rewritten across the nation without becoming accessible to the majority of users. The methodology employed must be such that the courseware developed can be transported across machines currently in use and future machines. This approach is being used in traditional non-graphics software. However, the graphics of microcomputers require new and different strategies. Development and testing of these strategies is being undertaken by Carroll College and the Waukesha Public School System under a National Science Foundation Grant.

The Educational Community of today varies widely in its computer knowledge and expertise. There exists a large number of users who have little knowledge of computers or the use of computers. However, for a number of years, they have been receiving an ever-increasing amount of "pressure" to use computers in their teaching. This situation is especially true in the areas of the sciences and mathematics. It is impossible to attend a conference in math or science without a barrage of presentations on the author's favorite use of computers. Many of these are excellent applications of computer usage, but require a great deal of computer expertise on the part of the presenter and often also the audience.

Many educators would like to make use of computer technology in their courses, but do not have the expertise or time to develop their own software. Much software is being written today, but the quality and ease of use of this software often leaves much to be desired. Assuming that quality software does exist for a particular course, it is often very machine-dependent. This will necessitate a difficult translation, and some applications will have to be re-written in order to transport the software to other hardware. This transportation is often a very costly and error prone process. Often transportability is ignored completely in the initial programming, and this may make the cost of transportation to other systems prohibitive.

One of the problems that must be of major concern to Computer Scientists is the ever-increasing cost of software. Software has long ago passed hardware as the major portion of cost in a computer system. Software specialists have been striving for years to reduce the costs associated with computer software. The area of structured design and structured programming has proven itself as a method for reducing initial costs and maintenance costs. However, even with these techniques, the costs of writing and re-writing software every time a new machine is purchased can no longer be tolerated. This is especially true in education where declining enrollments are greatly reducing funds available for developing software. In primary and secondary schools, funds just are not available to develop software. Teachers here must handle their own computer software. With the advent of microcomputers, most teachers have or will have a computer available for their use. However, these small computers require a great deal of expertise to develop software for them. Anyone who has read the documentation and PEEKed or POKEd their way through a complex program will attest to this statement. Therefore, quality software must be shared among the educational community. User groups have been set up to facilitate this sharing. However, these groups are usually tied to specific hardware and programs must be changed to run on other machines. With the advances being made in microcomputer technology, software must be somehow transportable across machines. Due to the low cost and variety of microcomputers, the teacher may not know what the capabilities of next year's machine will be. School administrators and board members often decide what hardware will be available and their decisions vary from year to year. This means that courseware will have to be transportable across microcomputers.

---

[1] Some of the materials incorporated in this work were developed with the financial support of the National Science Foundation Grant #SED80-12457. However, any opinions, findings and conclusions or recommendations are those of the author and do not necessarily reflect the views of the Foundation.

It has been demonstrated that educational software may be successfully transported across many different computers. CONDUIT at the University of Iowa(Hepler, 1980) and the CADA Research Group at the University of Iowa (Isaacs and Novick, 1978) have distibuted educational software for years. However, until recently this software has been restricted mainly to computers other than microcomputers, and to textual software rather than graphic software. CONDUIT is now distributing some of its software on the TRS-80, Commodore Pet, and Apple II. However, these packages must be rewritten for each machine type and CONDUIT is just beginning to use the graphics features of the microcomputers in its packages. Due to the differences of the graphic implementations on the various microcomputers, CONDUIT must expend much more time and effort on these packages.

A microcomputer that has graphic capability will be assumed to have line drawing capability and what I will call high resolution graphics. High resolution graphics is at least a resolution of a grid of 100 by 100 individually addressable points or pixels. Another feature that we will consider is color. For our particular application, it is felt that color will allow us to present concepts that would be much more difficult to convey using black and white.

To obtain a truly transportable color graphics system, the following major difficulties had to be overcome:

Hardware
Computer Language
Graphics Language
Graphics Resolution
Handling Color
Output Device
Handling Future Hardware

It is possible to design a system that takes care of the above so that the courseware developer need not be concerned with the above details.

## HARDWARE

A minimum of hardware restrictions was maintained throughout the systems design. However, it became necessary to impose several hardware restrictions. At the outset, it was decided that microcomputers did not have powerful enough editors to enable ease of entry and modification of courseware. Secondly, due to the lack of utilities and the slow speed of microcomputers, it was necessary to insist upon development on a larger machine. Therefore, the power of the large machine will make creation of the courseware much easier. However, this requires that the microcomputer must have the ability to communicate with the larger host machine and the ability to upload and download files. This capability exists on most microcomputers as an option, although a special communication software package must be found or created.

Since courseware often takes large amounts of memory, it must be modular in design. This modular design will allow the limited memory of the microcomputer to handle the execution of the smaller pieces of the courseware. This means that each module is going to have to load and execute another module upon completion. Since most modules may have several different terminations, a random access capability must also be present so that different modules may be loaded quickly. At this time, this capability on microcomputers is being handled by a floppy disk or hard disk auxiliary storage system.

Each module must be of sufficient size, so that the courseware does not spend most of its time loading new modules. For our applications, it has been found that a minimum of 24K bytes of memory be available for each module. This does not include memory for the microcomputer's operating system, language system, graphics storage, and graphic character sets. On an Apple II, this amounts to 24K bytes of memory for these necessary system features and 24K bytes of memory for application programs.

Since we are insisting on using color graphics, these features must be present and usable. The color requirement of at least four colors displayable at one time has been decided to be minimal, along with the graphics resolution of at least 100 by 100 addressable points. The ability to draw a line between two points is also required. Due to the low resolution, not all graphs will appear exactly as desired. There may be gaps or inconsistencies in output of some lines. However, it must be kept in mind that resolution is being sacrificed for price. This resolution of graphics is not sufficient for CAD/CAM or other such applications, but is adequate to give an approximate visual perspective.

## COMPUTER LANGUAGE

The next area to be decided is the language to be used as the base language for courseware creation. Since it was necessary to select a language that was implemented and would be implemented across a wide variety of microcomputers, it was felt that PASCAL and BASIC were the two choices. The criteria used to determine which language would be used were: ease of learning, ease of use, built-in structure, size of established user base, and capabilities within the language for output and graphics. It was felt that only in terms of structure did PASCAL have an advantage. Therefore, BASIC was chosen as the design language. Since structure is considered very important, the BASIC selected for the language in which courseware was to be written is the proposed ANSI Standard Basic. This BASIC has structure built into it. DO loops, EXIT conditions, SELECT, CASE, and IF blocks are all a part of this BASIC. Graphics are also implemented by SET VIEWPORT, SET WINDOW, PLOT, CLEAR, and other statements. The proposed standard makes a powerful tool, and since it is a natural extension of the BASIC known by an extremely large user base, it will make a good courseware development language.

Since ANSI BASIC does not exist, the educator will create a module on the host machine and compile it by means of a preprocessor that will check for syntax and make any necessary modifications to create an object code for the target machine. This object code will then be downloaded to the target machine where it can be tested. This process will be implemented by a single command and spare the developer the need to learn many commands on the host machines.

## GRAPHICS LANGUAGE

Besides the graphics capabilities built into the computer language, a large number of graphics tools are needed to allow the courseware developer to use a routine instead of reinventing complicated software. This approach is used in such often used packages as the International Mathematics Statistics Library and and the Plot-10 Software of Digital Equipment Corporation. Our graphic routines have been implemented by either BASIC or assembly language routines on the target machines. The developer only has to use the BASIC CALL statement to access the routine. The ANSI CALL statement will then be modified as necessary and the desired routine loaded automatically for the developer. These routines give the courseware developer a great deal of power. Routines such as BACKGROUND, BORDER, AXIS, SCREEN, FONT, UNDERLINE, PROMPT, GETCHAR, and MEASURE are now available. This list will naturally increase as the NSF project progresses. This will give the developer a large vocabulary to use.

One of the problems encountered is "How should graphics text be handled?" Since graphics text has to have its own symbol table or other method to display it, graphics text must be handled differently on each machine. Three different size character sets are created for the letters, special characters, and other graphic characters (pi, epsilon, square root, absolute value, etc.) and displayed according to ASCII position. However, the developer need not be concerned with those details. There exists a GRAPHIC PRINT statement that displays the desired text graphically. To display one of the graphic characters, the developer surrounds the keyword (PI, ANGLE, ERASE, etc.) by a pair of #'s in the GRAPHIC PRINT list.

## GRAPHICS RESOLUTION

Since the different microcomputers will have a wide variety of graphics resolutions, the resolution must also be handled by the system and not by the courseware developer. The method selected was to implement a configuration data file that contains descriptions of the desired machine in terms of horizontal and vertical dimensions. This file is created at system generation and read by the modules into variables that are known to the courseware developer. The developer will now code in terms of these variables and not in terms of actual screen coordinates. Actually, the only time that these variables are considered is in terms of the SET VIEWPORT. The actual positioning and plotting in the viewport will take place in terms of the developer's problem coordinates which are determined with a SET WINDOW statement. This means that the developers do not have to translate their thinking from their problem coordinates to a particular screen. All PLOTs and SET POSITIONs are in terms of problem coordinates.

## HANDLING COLOR

Color also has to be handled in a

special way, because microcomputers do not implement their color in the same way. Apple has six different colors, but you may only specify a certain color in certain columns. Since the developer is plotting in problem coordinates, there is no way to determine if a point is to be in the correct column. All of the graphic statements have code built into their routines to handle the plotting of color in the correct column. The routines will move the point into a correct column and then perform the desired operation. Since the graphics being implemented on these microcomputers is not being used for highly accurate plots, this technique will provide a good approximation to the desired plot.

Since not all machines implement the same colors, the developer may not select actual colors. The developer uses color variables that have values corresponding to the various colors. These can then be changed from machine to machine and the various graphic routines will take the color from the variable into account.

## OUTPUT DEVICE

Even the output device of the actual user of the courseware must be taken into account. The various monitors and televisions do not have the same ratio of vertical distance between points to horizontal distance between points. This means that a plot will not look the same on the different devices. For example, a circle on one device may appear as an ellipse on another. Therefore, all references to viewport will automatically transform it by the appropriate ratio. In this way, the developer need not worry about the output device. For each different output device, the ratio is established but once at system generation. This ratio also resides in the configuration file. At system generation, a program is run interactively that allows the ratio to be changed until the user is happy with the output for that device. The ratio is then automatically inserted into the configuration file. For different output devices, the user only need execute the system generation program.

## HANDLING FUTURE HARDWARE

One of the major problems encountered with courseware is the introduction of a new computer system. This usually necessitates a lengthy translation at best and often a total rewrite. However, using the above procedure, the source code need not be changed. Any necessary modifications will be made to the preprocessor and its associated routines.

Using the above high level graphics language defined as a subset of the proposed ANSI Standard BASIC(See Appendix A), the Graphic Extension Routines (See Appendix B),and Miscellaneous Routines (See Appendix C), the courseware developer can easily create a graphics system that can be automatically transported across machines. The creation of the initial courseware has proved to be much easier and faster for the inexperienced developer using this methodology. A minimum of experience with the BASIC programming language and a short tutorial in the specifics of our graphics language is all that is necessary. It is important that the actual users of the courseware be involved in the development of their courseware. They must have a tool such as this graphics language to aid in the courseware development.

## References

Hepler, Molly L. Pipeline, 5, No. 2. Iowa City: CONDUIT, 1980.

Isaacs, Gerald L. and Novick, Melvin R. Manual for the Computer Assisted Data Analysis Monitor. Iowa City: The University of Iowa, 1978.

## APPENDIX A

Allowable ANSI statements

| Statement | Restriction |
|---|---|
| REM | None |
| LET | Must supply keyword LET |
| FOR | None |
| NEXT | None |
| INPUT | None |
| READ | None |
| DATA | None |
| RESTORE | None |
| GOTO | None |
| GOSUB | None |
| RETURN | None |
| STOP | None |
| END | None |
| PRINT | None |
| DO | May not use while or until modifiers |
| EXIT IF | None |
| LOOP | None |
| CHAIN | None |
| ON GOTO/GOSUB | None |
| DIM | None |
| IF | Only allows "IF (cond) THEN line-number"<br>or<br>"IF (cond) THEN line-number ELSE line-number"<br>or<br>"IF (cond) THEN"<br>Allows IF block structure provided nothing follows THEN,ELSE,END-IF on their respective |

| | |
|---|---|
| | lines |
| ELSE | None |
| END IF | None |
| DEF FN | Single line, single letter, single parameter |
| SELECT | String or numeric arguments |
| CASE | Single relation or single TO-interval no lists or single constants |
| CASE ELSE | None |
| END SELECT | None |
| RANDOMIZE | None |
| SET VIEWPORT | In terms of screen variables |
| SET WINDOW | None |
| SET LINESTYLE | In terms of color variables |
| PLOT | None |
| SET POSITION | None |
| CLEAR | None |
| GRAPHIC PRINT | List is restricted to string expressions |

## APPENDIX B

### Graphic Extension Routines Implemented via CALL Statement

AXIS
BACKGROUND
BORDER
GRAPHICS
SCREEN
FONT
GRAPHIC INPUT
PROMPT
GETCHAR
MEASURE
UNDERLINE

## APPENDIX C

### Miscellaneous Routines Implemented via CALL Statement

TEXT
CLEAR END SCREEN
CLEAR END LINE
CLEAR TEXT
CONTINUE
ILLEGAL
REVERSE
PARSER
INITIALIZE
NUMERIC
OPEN
CLOSE
READ
WRITE

# SOURCES FOR AND EVALUATION OF K-12 MICROCOMPUTER SOFTWARE

Dorothy H. Judd
1990 Flagstaff Court
Glendale Heights, IL, 60137
Northern Illinois University

## Abstract

This paper reports on the survey results of a questionnaire sent to selected Illinois teachers in the fall of 1980 who were reported as using microcomputers in their classroom teaching. The focus of the present report is limited to software source awareness and teacher evaluation of alternative sources for software that is useful in the classroom. It is clear that much needs to be done to provide a greater awareness of software availability.

The idea that microcomputers will become a major factor in elementary and secondary education has been commonly acknowledged by educational leaders. However, the unavailability of effective software is an often heard complaint. This paper reports on the first survey research in Illinois addressed to discovering teacher's knowledge of software sources, and opinions on the effectiveness of specific software sources.

The results of a survey of 197 teachers in selected Illinois schools and their experiences with software sources for use with microcomputers are reported. More than sixty percent responded with answers to one part or another of the survey instrument in October and November, 1980. The questionnaire used in the study asked teachers to respond as to the extent of software use in the classroom. Microcomputer vendors, software program vendors, magazine sources, other teachers, district or a higher level of service agency, and their own programming efforts were rated. Teachers who responded that they created their own software programs were asked to report in what computer language they wrote such programs. All teachers in the study were asked to report on whether they used an 'authoring' language such as TUTOR or PILOT, and, if so, which language? They were also asked to report on whether they used an 'authoring system' such as GENIS or MicroGnome CAIware, and, if so, which system?

The final section of the survey instrument asked teachers to rate the effectiveness of software from the several sources available to them. The rating scale called for one of four answers:

most effective source in your classroom use

usually effective in classroom use

not effective in classroom use

no opinion or knowledge.

The experiences reported in the study on which this paper is based show a variety of sources and opinions revealed in the tables that follow. Provision was made in the survey for comment by respondents and more than a dozen teachers added comments regarding software sources and use of software.

One of the outcomes of the study is the realization that software evaluation is a significant and necessary element in education's use of microcomputers in the classroom. In this respect, microcomputer software is no different from textbooks and the textbook evaluation that has been a familiar part of elementary and secondary curriculum decisions for years. Moreover, it is also clear from the study results that information about microcomputer software sources is needed.

An approach to needed information on microcomputer software is found in the recent work of Jostad and Kosel[1]. Their article in the AEDS Monitor reported on the evaluation of more than 1200 programs from 35 companies providing microcomputer software for school use. The Jostad and Kosel article should be required reading for any curriculum director considering the use of microcomputers. In addition to the publications of the Association for Educational Data Systems (AEDS) there has been the relatively recent development of such specialized magazines as The Computing Teacher, Educational Computer magazine, and Classroom Computer News[2] which provide a further solution to the often heard complaint of a lack of information. The problem of disseminating information about these magazine sources, themselves, still remains. To be useful to educators, an awareness of these magazines' existence and their inclusion of information about microcomputer software must become known.

One finding of the current study[3], that teachers like the software they personally create the best and use it slightly more often than they use software available from a hardware vendor is evident in Tables 1 and 1a. More than 70% of the respondents reported using their own software program creations frequently or occasionally, compared with the slightly lower rating they gave hardware vendor software as to the frequency of its use (see Tables 6 and 6a). A solid twelve percent reported that the hardware vendor software was "not effective in classroom use."

Table No. 1

Question: To what extent do you use software from the sources listed below?

Self (software you create)

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. Frequently | 51 | 44.0 % |
| 2. Occasionally | 31 | 26.7 |
| 3. Seldom | 14 | 12.1 |
| 4. Never | 7 | 6.0 |
| 5. No opinion or knowledge | 7 | 6.0 |
| No answer | 6 | 5.2 |
| Total | 116 | 100.0 % |

Mean = 1.982   Standard deviation = 1.196

Table No. 1a

Question: Your opinion on the effectiveness of software is important. Please circle the number on the right that identifies the answer for each software choice.

Self (software you create)

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. most effective source in your classroom use. | 43 | 37.1 % |
| 2. usually effective in classroom use. | 41 | 35.3 |
| 3. not effective in classroom use. | 2 | 1.7 |
| 4. no opinion or knowledge. | 22 | 19.0 |
| No answer | 8 | 6.9 |
| Total | 116 | 100.0 % |

Mean = 2.028   Standard deviation = 1.115

Tahle No. 2

Question: Do you create software? If Yes, in what computer language do you write it?

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| YES | 90 | 77.6 % |
| NO | 21 | 18.1 |
| No answer | 5 | 4.3 |
| Total | 116 | 100.0 % |

Of the 90 YES answers, languages named were:

BASIC - 78
BASIC and Assembly or machine language - 4
BASIC and FORTRAN - 3
BASIC and C-BASIC - 1
BASIC and PASCAL - 1
FORTRAN - 2
PL/1 (PL/C) - 1

In the face of the heavy percentage that reported writing their own software, an even larger percentage reported using BASIC. The number and variety of other languages reported by the teachers in this study is reported in Table 2. Those educators who have used authoring languages or authoring systems may be disappointed with the answers revealed in Tables 3, 4, 5). Clearly, the idea of using an authoring language is known to a very few (11.2%), while use of an authoring system such as GENIS or MicroGnome CAIware is known to even fewer (8.6%).

In the discussion above, mention was made of the role magazines can play in providing information about microcomputer software. At the time of the present study, one of the magazines mentioned earlier was not in existence, and it is the conclusion of the writer that little was known by the teacher respondents about the availability of either Classroom Computer News or The Computing Teacher. Thus, the answers reported in Tables 9 and 9a reflect the general microcomputer press. For all their general appeal and lack of special concern with education, the microcomputer magazines can take some pride in the degree to which they are reported as a source for software teachers use in the classroom. It is worth noting that only 10.3% feel that "magazine source" software is not effective in the classroom, compared with the 12.9% figure for hardware vendor software.

Table No. 3

Question: Have you used an "authoring" language such as 'TUTOR' or 'PILOT' to create programs for classroom use? If YES, in what language?

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| YES | 13 | 11.2 % |
| No | 97 | 83.6 |
| No answer | 6 | 5.2 |
| Total | 116 | 100.0 % |

Of the 13 YES answers, responses were:
PILOT - 10
PILOT and TUTOR - 1
TUTOR - 1
Plan on learning one of above.- 1

Table No. 4

Question: Your opinion on the effectiveness of software is important. Please circle the number on the right that identifies the answer for each software choice.
Authoring language 'PILOT'

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. most effective source in your classroom use. | 6 | 5.2 % |
| 2. usually effective in classroom use. | 5 | 4.3 |
| 3. not effective in classroom use. | 0 | 0.0 |
| 4. no opinion or knowledge. | 96 | 82.7 |
| No answer | 9 | 7.8 |
| Total | 116 | 100.0 % |

Mean = 3.626 Standard deviation = 0.906

Table No. 5

Question: Have you used an "authoring" system such as 'GENIS' with the Apple or 'MicroGnome CAIware' with the TRS-80? If YES, which one?

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| YES | 10 | 8.6 % |
| No | 101 | 87.1 |
| No answer | 5 | 4.3 |
| Total | 116 | 100.0 % |

Of the 10 YES answers, responses were:
GENIS - 7
MicroGnome CAIware - 3

The fellow teachers of those who responded to the current survey are an important source of software for classroom use. Other teachers represent almost as large a source as magazines, according to answers given in this study. The answers on this question are set forth in Tables 10 and 10a. A real puzzle in magnitude of response is found in Tables 11 and 11a. The software vendor is less often used as a source for classroom software than might have been expected and is not as well regarded as software sources reported on above.

Despite the success stories of the Minnesota Educational Computing Consortium (MECC), and other agencies in parts of Michigan, and in Iowa, the idea of a district level or higher service agency providing any help or information on microcomputer software for classroom use has not taken hold in Illinois, according to the data reported in Tables 12 and 12a. More than 77% of the respondents report no use or knowledge about district level service agency help on software. The educational establishment in Illinois

Table No. 6

Question: To what extent do you use software from the sources listed below?
Hardware vendor (Apple, Pet, Radio Shack, etc.)

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. Frequently | 32 | 27.6 % |
| 2. Occasionally | 49 | 42.2 |
| 3. Seldom | 17 | 14.7 |
| 4. Never | 7 | 6.0 |
| 5. No opinion or knowledge | 6 | 5.2 |
| No answer | 5 | 4.3 |
| Total | 116 | 100.0 % |

Mean = 2.153 Standard deviation = 1.080

Table No. 6a

Question: Your opinion on the effectiveness of software is important. Please circle the number on the right that identifies the answer for each software choice.
Self (software you create)

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. most effective source in your classroom use. | 32 | 27.6 % |
| 2. usually effective in classroom use. | 41 | 35.3 |
| 3. not effective in classroom use. | 15 | 12.9 |
| 4. no opinion or knowledge. | 22 | 19.0 |
| No answer | 5 | 5.2 |
| Total | 116 | 100.0 % |

Mean = 2.245 Standard deviation = 1.085

centers around local school districts that are part of an Educational Service Region (ESR). The current study indicates there is little that these ESR's are doing in terms of microcomputer software assistance to teachers in the classroom. One notable exception is found in the Career Education Center attached to the DuPage County ESR in Wheaton, Illinois where the director has pioneered training for and development of software useful in classrooms throughout the DuPage county school service area. Still another exception in Illinois has been the career education software service provided by a Career Education Center attached to Joliet Junior College in Joliet, Illinois. Probably there are other district agencies or regional service units that are engaging in assistance to the classroom teacher on microcomputer software needs, but they have not become known to the writer, nor are they reported in the survey instrument on which this paper is based.

Of interest to some may be the reported "other" sources found in Tables 13 and 13a. User groups, student created programs, and MECC programs each rate two mentions. Were the same questionnaire to be used in another year, it is to be expected that answers would change on a number of questions. For example, the Institute for Educational Research (IER) answer in Table 13 represents a disappointingly small recognition of an organization that is endeavoring to provide MECC type services in Illinois. This result is all the more unfortunate since the IER organization has enjoyed state funding as well as cooperative funding from a number of school districts. The whole issue of district or higher level agency service, let alone non-state agency service to classroom teachers seeking to use microcomputers in their curricular work in the classroom, needs major research and developmental effort in Illinois, and perhaps in a number of other states. If the promise that microcomputers represent to the instructional reponsibility of elementary and secondary schools is to be fulfilled, the issue of software availability and service by other than building leadership will need to be solved. Whether this is provided by district curriculum leadership, by educational service region offices, or by MECC type agencies, remains to be resolved in Illinois.

## Endnotes

[1] Karen Jostad and Marge Kosel, "Survey of Commercial Software," AEDS Monitor, 19:19-21, October-December, 1980.

[2] for subscription information contact The Computing Teacher, c/o Computing Center, Eastern Oregon State College, La Grande, OR, 97850; Classroom Computer News, Box 266, Cambridge, MA, 02138; and Educational Computer Magazine, Box 535, Cupertino, CA, 95015.

[3] those interested in the complete findings of the study referred to herein are welcome to inquire as to their availability from the author.

### Table No. 7

Question: To what extent do you use software from the sources listed below?

Magazine sources

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. Frequently | 18 | 15.5 % |
| 2. Occasionally | 39 | 33.6 |
| 3. Seldom | 24 | 20.7 |
| 4. Never | 19 | 7.8 |
| 5. No opinion or knowledge | 9 | 7.8 |
| No answer | 7 | 6.0 |
| Total | 116 | 100.0 % |

Mean = 2.651 Standard deviation = 1.189

### Table No. 7a

Question: Your opinion on the effectiveness of software is important. Please circle the number on the right that identifies the answer for each software choice.

Magazine sources

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. most effective source in your classroom use. | 10 | 8.6 % |
| 2. usually effective in classroom use. | 50 | 43.1 |
| 3. not effective in classroom use. | 12 | 10.3 |
| 4. no opinion or knowledge. | 33 | 28.5 |
| No answer | 11 | 9.5 |
| Total | 116 | 100.0 % |

Mean = 2.648 Standard deviation = 1.028

### Table No. 8

Question: To what extent do you use software from the sources listed below?

Other teachers

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. Frequently | 15 | 12.9 % |
| 2. Occasionally | 27 | 23.3 |
| 3. Seldom | 21 | 18.1 |
| 4. Never | 34 | 29.3 |
| 5. No opinion or knowledge | 13 | 11.2 |
| No answer | 6 | 5.2 |
| Total | 116 | 100.0 % |

Mean = 3.027 Standard deviation = 1.260

### Table No. 8a

Question: Your opinion on the effectiveness of software is important. Please circle the number on the right that identifies the answer for each software choice.

Other teachers

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. most effective source in your classroom use. | 19 | 16.4 % |
| 2. usually effective in classroom use. | 42 | 36.2 |
| 3. not effective in classroom use. | 8 | 6.9 |
| 4. no opinion or knowledge. | 39 | 33.6 |
| No answer | 8 | 6.9 |
| Total | 116 | 100.0 % |

Mean = 2.620 Standard deviation = 1.150

Table No. 9

Question: To what extent do you use software from the sources listed below?

Software vendor

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. Frequently | 15 | 12.9 % |
| 2. Occasionally | 31 | 26.7 |
| 3. Seldom | 24 | 20.7 |
| 4. Never | 30 | 25.9 |
| 5. No opinion or knowledge | 10 | 8.6 |
| No answer | 6 | 5.2 |
| Total | 116 | 100.0 % |

Mean = 2.900 Standard deviation = 1.211

Table No. 9a

Question: Your opinion on the effectiveness of software is important. Please circle the number on the right that identifies the answer for each software choice.

Software vendor

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. most effective source in your classroom use. | 10 | 8.6 % |
| 2. usually effective in classroom use. | 37 | 31.9 |
| 3. not effective in classroom use. | 12 | 10.3 |
| 4. no opinion or knowledge. | 48 | 41.4 |
| No answer | 9 | 7.8 |
| Total | 116 | 100.0 % |

Mean = 2.916 Standard deviation = 1.083

Table No. 10

Question: To what extent do you use software from the sources listed below?

District level (or higher) service agency

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. Frequently | 3 | 2.6 % |
| 2. Occasionally | 13 | 11.2 |
| 3. Seldom | 10 | 8.6 |
| 4. Never | 58 | 50.0 |
| 5. No opinion or knowledge | 25 | 21.6 |
| No answer | 7 | 6.0 |
| Total | 116 | 100.0 % |

Mean = 3.817 Standard deviation = 1.011

Table No. 10a

Question: Your opinion on the effectiveness of software is important. Please circle the number on the right that identifies the answer for each software choice.

District level (or higher) service agency

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. most effective source in your classroom use. | 6 | 5.2 % |
| 2. usually effective in classroom use. | 15 | 12.9 |
| 3. not effective in classroom use. | 8 | 6.9 |
| 4. no opinion or knowledge. | 76 | 65.5 |
| No answer | 11 | 9.5 |
| Total | 116 | 100.0 % |

Mean = 3.467 Standard deviation = 0.941

### Table No. 11

Question: To what extent do you use software from the sources listed below?
Other (please specify)

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. Frequently | 4 | 3.4 % |
| 2. Occasionally | 4 | 3.4 |
| 3. Seldom | 0 | 0.0 |
| 4. Never | 9 | 7.8 |
| 5. No opinion or knowledge | 19 | 16.4 |
| No answer | 80 | 69.0 |
| Total | 116 | 100.0 % |

Of eight answers, responses were:
student created - 2 user groups - 2
MECC - 2 IER - 1
friends who have microcomputer - 1

### Table No. 11a

Question: Your opinion on the effectiveness of software is important. Please circle the number on the right that identifies the answer for each software choice.
Other (please specify)

| Answer Choice N = 116 | Number | Percent |
|---|---|---|
| 1. most effective source in your classroom use. | 3 | 2.6 % |
| 2. usually effective in classroom use. | 0 | 0.0 |
| 3. not effective in classroom use. | 1 | 0.8 |
| 4. no opinion or knowledge. | 32 | 27.6 |
| No answer | 80 | 69.0 |
| Total | 116 | 100.0 % |

Of the 3 answers above there was one mention each of user groups, MECC, and IER.

### Table 12

Some Questionnaire Comments Received:

Suburban elementary teacher

"Hesitate to order software by mail."

*

Suburban elementary teacher

*

"This is our first year with the computer and we are using some Milliken Math material as well as some teacher made things. It is slow, but we are encouraged."

*

Downstate elementary teacher

"Please send sources of getting software."

*

A junior high school math teacher:

"I hope the goal of this survey is to get the 'educational' ball rolling. There needs to be a network of sharing good educational software."

*

Junior high school teacher

"I hope this is not only a survey. I would like to get help in various areas and am anxious that you help me in these, like 'Pilot' and GENIS."

*

A junior high school math teacher:

"We recently completed a 2 and 1/2 week session with PILOT. Every student was able to type in and run at least four programs of the 15 programs they wrote. The basic R; S; D; T; A; J; E; and C; commands were used concluding the very successful unit with problem solving. Five TRS-80's were used with printers."

*

A high school teacher:

"I enjoy teaching microcomputers. Students enjoy a new challenge. Wish other teachers did not feel so threatened by them. Will be glad when more material is available."

*

A high school teacher:

"Finding personal time to practice and develop programs for class use is a prolem - Students have priority on the one computer we have now. Plans call for more computers with at least one portable moving into classrooms in 81-2 or later."

*

A downstate high school teacher:

"I need more contact with others interested and using computers in the classroom, to share and exchange ideas and software. Need some kind of newsletter."

# THE STATE'S ROLE IN EDUCATIONAL COMPUTING:
# VIEW FROM THE LEGISLATURE

Phyllis Kahn
237 State Office Building
St. Paul, Minnesota 55155
House of Representatives

For the past decade, the Minnesota Legislature has taken an active interest in the implementation of a coherent plan for educational data processing. The decisions that have been made during this time cover a wide range, including the structure of the decision process of educational computing systems, the concept of centralized versus decentralized hardware, the nature of state financial support and even some of the finer points of hardware acquisition. The questions of how to obtain equity in the area of computer access in this diverse state and how to make good yet neutral judgments have been prominent problems.

My interest in the use of computers in state government is a part of the interest I have had in general in technological public policy issues and is related to my professional training as a scientist before being elected to the legislature in 1972.

One introductory comment is that elected officials in general have a disappointing record in the area of legislating scientific and technological issues. Part of the reason is that these issues often require a long lead time to come to a meaningful conclusion and to many elected officials anything longer than the next election (two years or so away) might as well be infinity. Additionally technology working right is not the stuff exciting political careers are made of, so it becomes hard to ask political types to put in long difficult hours on a subject with few obvious public rewards.

My experience in this area has come as the chair of a subcommittee on computers which is part of the House of Representatives Appropriations Committee. This subcommittee was formed because of the observation of dramatically increasing budgets for both state government and educational systems with the following characteristics:

-There appeared to be little critical oversight from either the legislature or the Governor's office. Department budgets in this area were set and hardly changed as they moved through the process.

-The State of Minnesota is a model of a heavily centralized system with one agency controlling state departments and another educational institutions and school districts.

-There appeared to be recurring turf problems, with a lack of separation of policy formation, operation and critical overview. For example, the decision to computerize (or how to) was usually made by the centralized computer agency itself.

-There appeared to be acquisition problems with the state bidding system. A bid system emphasizing the lowest initial cost may be adequate for typewriters or trucks, but will fall down on a complicated purchase such as computers.

-The political need for equal access to computer usage for both metropolitan areas and outstate districts leading to the state subsidy of the telecommunications network.

-We also observed conflict of interest problems with public officials sometimes allied with private firms doing business with the state, leaving an uncomfortable appearance of entangling alliances.

-Finally how could the state best take advantage of the new directions of the computer industry with the trend to micro and mini computers without returning to an era of uncontrolled proliferation.

The specific legislative interest in educational computing started in 1971 with the Minnesota Miracle, the process of great infusion of state funds into the previously locally-funded school districts. Along with this funding came a need to track and understand how the money was being spent.

A first step in 1971 was the passage of a law requiring school districts to submit annual plans and budgets with the State Department of Education and to obtain departmental approval before expending funds for computer activities.

In 1973 MECC, the Minnesota Educational Computing Consortium, was formed as a point powers agreement by the State Department of Education, the University of Minnesota, the State University System and the Community College System to provide computer services to its constituent members. In 1973 and 1975 the legislature started to pass on the financial arrangements for MECC and monitor its progress.

In 1976, the original UFARS (Uniform Financial Accounting and Reporting Standards) legislation was passed requiring all districts to convert to computerized accounting systems, effective July 1, 1980.

In addition in 1976 the legislature expressed disatisfaction with MECC's intent to purchase a Univac 1110 for instructional time sharing and insisted on a consultant-monitored reevaluation of that contract, resulting in a cancellation of that contract.

In 1979 and 1980 the legislature took an active interest in restructuring the Educational Management Information System. Questions had been raised concerning this system's cost effectiveness, quality of software, hardware purchase decisions, sensitivity and responsiveness to local user requirements and needs.

A consultant was retained to perform the evaluation and answer specific questions, including the comparison of centralized and decentralized systems; the state level of funding and organization; the formation, function and structure of regional centers including the assignment of districts to regions and an analysis of hardware and software decisions, development and maintenance.

The major recommendations of the consultant, (Peat, Marwick Mitchell and Company) are now in the process of being implemented. A major thrust of their recommendations has been that the state avoid a single, totally uniform and monolithic system, but instead adopt common system specifications allowing differences between school districts with simple, intermediate or complex needs for management information.

Additional aspects of their report now in the process of implementation include:

-A requirement for a meaningful systems architecture and a long range plan,

-Clarification of an ambiguous decision structure. This has been accomplished by removing the authority for policy setting and planning from MECC and placing it in the State Department of Education.

-Establishment of the E.S.V. (Elementary, Secondary, Vocational) Computer Council to make judgment calls in a manner somewhat independent of the operating agencies. Among the duties of the Council are to advise and assist the State Board in the following areas: development of the systems architecture and the long range plan; the creation and alteration of regional management centers; the approval of the use by districts of alternative management information systems and the review of proposed data standards.

We intend that the State of Minnesota will meet its initial goals of providing timely, accurate and comparable information to legislators and others concerned with financial management of education. In addition, we hope to encourage flexibility, innovation and technical change with the objective of delivering cost-effective solutions to meet these needs.

# LOW COST COMPUTER/VIDEODISC LEARNING STATIONS

Kent T. Kehrberg
and
Richard A. Pollak
Minnesota Educational Computing Consortium
St. Paul, MN 55113
(612) 376-1111

The combination of microcomputers and videodisc players offers the possibility for a low cost learning station uniting the interactive strengths of computers with the audio, video, and animation features of television. Courses designed for these learning stations can help meet the objective of offering a variety of classes in schools with declining enrollments. A prototype economics course has been developed for individualized high school social studies instruction.

Both computer and television have long been recognized as tools to assist learning. Television with its audio, pictorial, and animation abilities has been used in education for over three decades. However, the medium is basically passive in nature. The classroom use of computers has been characterized by active participation on the part of students. This interaction, however, occurs primarily with text or simple graphic forms. Merging these two media into a single learning station unites the audio-video strengths of television with the interactive powers of the computer.

In the past, projects such as CAVE[1] have successfully merged these two technologies but with the high costs of a mainframe computer system, videotape player, and a custom made interface.[2] The advent of microcomputers has solved many of the problems associated with computers in video-enhanced instructional lessons. Their low cost, portability and nonreliance on telecommunications have provided a cost effective delivery system for individualized instruction. Improvements in disk storage devices will soon solve the mass storage limitation on microcomputers and eliminate the need to shuffle diskettes during the delivery of CAI.

Recent developments in video technology have produced the low cost consumer videodisc player. These units play a disc similar in size and shape to a phonograph record. Both video and audio information is recorded on the surface. Models available in the consumer market, such as Pioneer VP-1000, retail for $700 and offer such features as high speed searching, still frames, and slow and fast motion. The built-in remote control interface can be easily coupled with a microcomputer. The videodisc player and interface typically cost less than half the cost of a comparable videotape player and interface board.

The Special Projects Division of the Minnesota Educational Computing Consortium has been developing a model course using videodisc supported computer instruction during the past 18 months. The project is entitled "Personal Computers and the Home Videodisc Player: Solving Instruction Delivery Problems Created by Declining Enrollment."

The purpose of the project is to demonstrate that low cost learning stations comprised of a microcomputer and a consumer videodisc player can be programmed to deliver complete courses of instruction. The learning station consists of four integrated components totalling under $3,000. The components are an APPLE II microcomputer, Pioneer videodisc player, two television monitors and an APPLE II/videodisc controller (commercially available at a cost of $250).

Because the project calls for the use of a low cost microcomputer and MECC, through a statewide contract, has sold over 2000 APPLE II microcomputers, a decision was made to develop on these machines. These units typically cost around $1800. They are configured with the operating system APPLESOFT BASIC in ROM, 48K of RAM, one floppy disk unit which can store up to 140K bytes, and a small nine-inch black and white monitor. The videodisc player presently being used is the Pioneer Laser Disc model VP1000. The cost of the player is about $700. Along with the videodisc player, a 13-inch color monitor is used to view the video material.

The videodisc can display conventional video segments with related frame number references. The APPLE II, through the inexpensive commercially available interface, automatically locates and plays specific segments. It is also able to choose which audio track to play or combine them to produce stereo. This integrated combination of computer controlled videodisc provides a learning station that is able to take advantage of the audio-pictorial capabilities of educational television, the interactive ability of computers and the random accessibility of videodiscs.

The curriculum has been designed for an individualized nine-week high school economics course. The course is divided into five units with each of the units containing 8-12 learning sessions. These sessions are defined as the minimum amount of material that the learner should be able to cover in "one sitting" at the learning station. In general, the session takes between 15 and 30 minutes.

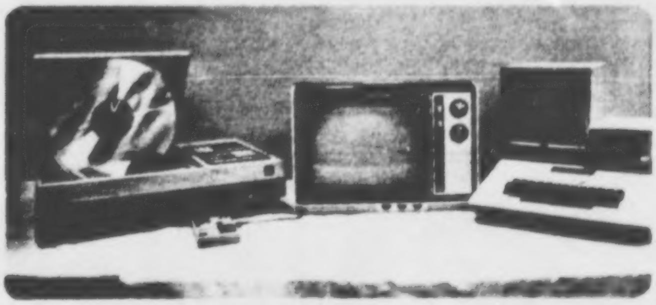
The need in Minnesota's schools seems to be increasing for this type of technology. With the declining enrollments in today's schools, rural and urban schools in particular are not able to offer a wide variety of classes to students. Therefore, the learning station and the instructional materials have been designed to be used in a school where there may be few students desiring to take economics and/or limited teacher resources. The instruction is designed to require a minimal amount of teacher assistance. Since the student materials are an individualized instruction package and could well be part of a media center in the school, the amount of time and role of the teacher is up to the individual controlling the learning activity. A teacher may meet with the student frequently or not, individually or in small groups.

Students in an individualized setting have occasionally been referred to as "lonely learners." Although the student still works alone at the learning station, every effort has been made to create a responsive environment. Each session begins with the student manual. After providing introductory statements and the objectives for the day's lesson, the student is directed to turn to the microcomputer. The microcomputer provides a few minutes of instruction and then in turn directs the student back to the manual or toward the television screen. Tutorial instruction, examples, reinforcement and even evaluation may be accessed from the videodisc player. These visual segments range up to three to four minutes in length. Using the microcomputer, videodisc or manual for only a few minutes at a time creates an active twenty-minute learning session.

The microcomputer and videodisc offers a low cost learning station. However, as with all forms of technology, they are useless without courseware. To date there is no commercially widely distributed educational courseware available for use with videodisc players. The American Broadcasting Company (ABC) and the National Education Association (NEA) are teaming up to produce a series of discs for elementary school use. However, these are being designed for use in a traditional classroom setting and not for individualized learning stations.

There are two immediate drawbacks to the development of courseware for videodiscs. First, the high cost of development. These costs exceed those of traditional CAI in as much as the need to produce video material also becomes a factor. Secondly, the absence of a market for a courseware product. Although microcomputers are becoming more common in schools, it is extremely rare to find a school with a videodisc player. Schools are reluctant to purchase equipment for which there is no software while the publisher is reluctant to develop software for which there are no customers. This chicken and egg phenomenon will be resolved only by a daring entrepreneur or significant public funding.

The cost of producing a videodisc itself does not pose a significant problem. The lower cost of videodisc copies when compared to videotape copies quickly reduces the initially high



Low Cost Computer/Videodisc Learning Station cost of preparing a master videodisc. Based on our experience, the break even point for tape versus disc is approximately 150 copies. Therefore, any large school district or regional consortia would find discs economical when exeeding this number of units. In addition, the significantly longer life of a disc, as opposed to tape, offers additional savings over a period of time.

Today's technology can provide low cost learning stations for education. Microcomputers, when coupled with videodisc players, can deliver an interactive visual based learning station for under $3000. These stations hold promise for the delivery of curriculum material to schools which are reducing course offerings due to low student/teacher ratios.

## References

1 Richard Pollak. "Computer Augmented Video Education (CAVE): An Overview," 9th Annual Conference on Computers in Undergraduate Curricula (CCUC/9), 1978.

2 Richard Pollak. "Controlling for the Computer Video Environment: A Computer Augmented Video Educational Experience," Annual Symposium of the Association for the Development of Computer Based Instructional Systems (ADCIS), 1980.

# THE GIS OPTIONS PROGRAM:
# A DYNAMIC APPROACH TO PROVIDING
# LOCALIZED CAREER INFORMATION

Mr. Gary King and Ms. Linda Kobylarz
Connecticut SOICC, and Time Share Corporation
Hartford, Connecticut

During the mid 1970's federal and state legislation initiated programs which were designed to support the development of state-wide career information delivery systems. Connecticut was one of the states so funded and selected the GIS Options Program as the basis for its computerized delivery system. The GIS Options approach provides users access to a system that includes national career information, as well as locally relevant information about occupations, vocational training schools, sources of financial aid, local business and industry etc. Comprehensive data bases, methods and controls in information gathering, programs to implement user services, in-service training, standard programing for a variety of computer systems, as well as timely delivery have made the GIS Options Program successful in Connecticut and 14 other states.

## Background

It has long been recognized that the availability of good information is critical to effective decision-making at various stages of the career development process. In recognition of this need, the Education Amendments of 1976, and the CETA Amendments of 1978 established the National Occupational Information Coordinating Committee (NOICC). A major responsibility of the NOICC is to assist states in the development and dissemination of locally relevant career information and to improve communication, coordination, and cooperation in the use of the system by which the information is made available to the public. To that end, grants were let by NOICC to several states in December of 1979. Under that program the Connecticut State Occupational Information Coordinating Committee (SOICC), established by executive order of the Governor in 1978, received funding to initiate a Career Information Delivery System (CIDS).

A major purpose of the grant that the Connecticut SOICC received from the federal government was to implement a state-wide computerized Career Information Delivery System (CIDS) which would develop and use relevant labor market information to help people in the process of career planning and job seeking. The Connecticut proposal, which was ranked number one in the country, presented an integrated approach to the implementation of the CIDS objectives. Under the grant guidelines, $300,000 of federal funds were made available to Connecticut over a two-year period. Numerous state and local agencies matched that amount, thus providing a total of $600,000 for the project's delivery. The degree of interagency cooperation is exemplified by the composition of the Connecticut SOICC as described below.

### Statutory Members

1. Commissioner, Connecticut State Department of Labor; representing the State Employment and Training Council.

2. Associate Commissioner for Vocational Education, State Department of Education; representing the State Vocational Education Agency.

3. Executive Director, Employment Security Division, Connecticut State Department of Labor; representing the State Employment Security Agency.

4. Assistant Director, Division of Vocational Rehabilitation, State Department of Education; representing the State Vocational Rehabilitation Agency.

### Non-Statutory Members

5. Commissioner, Connecticut Board of Higher Education.

6. Undersecretary for Employment and Training, Connecticut State Office of Policy & Management.

There is also a Technical Steering Group (TSG) which provides technical advice and guidance to the SOICC and the Executive Director. The membership of the TSG is determined and appointed by the SOICC and limited to twelve members.

The goal of the Connecticut CIDS is to have a positive influence on the process by which people choose jobs by providing them with current, accurate, and locally relevant occupational, educational, and training information. The availability of such information should help individuals make smoother transitions at key points during their career life, such as the transitions from school to work or return to the labor force.

Specific objectives of the Connecticut CIDS are to:

1. help students and clients learn about and understand the range of career opportunities presently available and those that are likely to be available in the future;
2. help entrants to the labor force become aware of occupations they would find acceptable and personally satisfying;
3. encourage persons in the process of career exploration and decision-making to seek out vocational information on their own;
4. increase awareness of major sources of occupational, educational and training information;
5. help people learn of educational and training opportunities and their relationship to occupations they may be exploring;
6. provide support for related programs, including career education, career and employment counseling, employment and training and educational planning.

## Types of Users

The CIDS project was charged with the responsibility of meeting the needs of a very diverse group of users as indicated below:

1. Students in Secondary Schools, Vocational-Technical Schools, Community Colleges, State Technical Colleges, and 4 Year Colleges
2. Employment Service Clients
3. Manpower Trainees (CETA)
4. Welfare Clients (WIN)
5. Vocational-Rehabilitation Clients
6. Corrections Institutions Inmates
7. Veterans
8. Adults in general

## System Selection

Many criteria were considered in selecting a software package to serve as the basic delivery vehicle for the Connecticut CIDS.

1. Demonstrated effectiveness with populations similar to those mentioned above.
2. Ease of use, interactive capabilities
3. Comprehensive national data base
4. Capability of adding local data bases
5. Technical excellence
6. Transportability to various hardware configurations
7. Availability of support in technical, information development, and user services areas

After careful review of several commercially produced software packages, The Guidance Information System (GIS) was selected as the CIDS information dissemination vehicle.

## Connecticut CIDS

### National Information Files

The Guidance Information System (GIS) is marketed by Time Share Corporation, a subsidiary of Houghton Mifflin Publishers. It contains national information in the following areas:

a. National Occupations (875)
b. Armed Services Occupations
c. Two-year Colleges (1500)
d. Four-year Colleges (1700)
e. Graduate Schools (1400)
f. National Financial Aids

The GIS is a very easy system to learn how to use. There are two approaches to getting the information. In one, the user simply asks the computer for a description of a specific occupation or school. In the other method, the user at the computer terminal employs a simple set of commands to enter characteristics such as personal interests, aptitudes, educational levels, or working conditions and can thus search for a possible list of occupations for consideration. The same process can be used to draw up a list of education or training institutions applicable to an individual's needs.

The GIS provides for great flexibility in that the user is free to start his/her search wherever he/she wants and to move through the system in a way dictated by his/her own personal priorities. The user is free to change his/her mind at any time to easily modify what has been entered into the computer. The user can also easily check to see why a paricular occupation or school was not part of his/her printout of qualifying options, thus facilitating further decision-making and self-assessment. The GIS is a very user oriented system. It does not make any assumptions about what course of action would be best for a user. It does, however, present a well organized display of the many characteristics a user might want to consider in his/her occupational exploration and then provides a means to easily related personal choices to occupations.

## Connecticut Information Files

Although the GIS provides comprehensive national data, it was felt that to truly meet the needs of Connecticut users both state and regional information needed to be added to the system. All information and analysis delivered by the CIDS is specifically tailored to meet user needs. Connecticut specific information includes: an Occupational Information File; a Vocational Training File; a Financial Aids Information File; and an Apprenticeship File.

## Connecticut CIDS Delivery Network

The Connecticut CIDS employs a host computer through which remote terminals tie into a PDP 11/70 which is strategically located in the State at New Haven. This computer network is supported and operated by the six Regional Education Service Centers (RESC's) of Connecticut, and has been in operation successfully and effectively since the early 1970's. The computer network is known as ESCCNET, the Educational Service Center Computer Network. The computer center and network is operated by the Area Cooperative Education Services (ACES), RESC. ESCCNET successfully has been delivering GIS national files to local agencies, statewide, but on a limited basis, for the past five years. During the course of the project, a total of 64 ports will be available for accessing the CIDS information. Through a mix of contention and dedicated line service, the system can adequately support more than 100 user sites.

Historically, the GIS/ESCCNET delivery system has been demonstrably effective with persons of varying ability and experience. The accessing and dissemination components used have enhanced the attractiveness of the information to individuals not only to draw them into using the system but also to stimulate and facilitate further occupational exploration. The expanded system supported and implemented under the auspices of the Connecticut CIDS consortium has continued to prove the effectiveness of this approach not only to past users but to a whole new array of user populations.

5

# Computerized Support Systems for Individual Educational Plans

T. Koehler, A. Ossler and S. Raucher
Montgomery County Public Schools
Rockville, Maryland 20850

ABSTRACT: Public Law 94-142, and the state bylaws implementing the law, has made the Individual Educational Plan (IEP) a major facet in the life of handicapped students and special education teachers and administrators. The requirement to plan annual objectives for special education students has resulted in elaborate and laborious procedures to create, review and negotiate these plans. The amount of typing, filing, recording, and teaching of IEP's has similarly created a significant clerical burden for all levels of school personnel.

In addition to the labor costs of IEP's the content presents a major challenge. The creation of valid and precise IEP's, that provide some measure of consistency across a school system is a nontrivial task. Associating valid objectives with instructional strategies, materials and assessment items presents a monumental manual task that is a prime candidate for automated assistance.

This paper presents a design and description of computer systems at Montgomery County Public Schools that support IEP's. The discussion covers the educational, technological and procedural issues as encountered since the project began in 1978. The evolution of the present system and plans for its future expansion are included.

## INTRODUCTION

The Individual Educational Plan (IEP) for special education students had enjoyed a limited vogue for a number of years. With the passage of the Federal P. L. 94-142, and the various state by-laws and regulations implementing the law, the IEP has become a major facet in the life of almost every special education teacher and administrator.

The IEP process is by necessity a laborious one. Much thought and attention is required to formulate precise objectives and other IEP content, and the preparation and typing of the plan is quite time consuming. The use of computer technology to reduce some of these labor costs is thus a fertile field.

The creation of a valid and precise IEP requires a great deal of conceptualization and definition of educational objectives. Formulation of these objectives is usually beyond the time available to most teachers, and often calls upon skills which are not necessarily within their training. Also, consistency of objectives within a school system can never be achieved if educational objectives are formulated ad hoc by every teacher.

There is, therefore, a valid need to develop means not only to reduce work associated with IEP's but also to improve quality. These two thrusts supply the basic requirements for any computerized IEP support system.

## THE MONTGOMERY COUNTY PUBLIC SCHOOL SYSTEM PROJECT - ORBIT INITIAL PROJECT

Based on the ideas described above, the Montgomery County Public School System (MCPS) began the initial work on a computerized IEP support system in 1978. Both the technological and educational issues were addressed. A preliminary analysis indicated that the development of a set of IEP instructional objectives was of first priority. Objectives, the foundation of any system, need to be specified prior to the technology being developed. In 1978, special education teacher specialists began the collection and refinement of varied educational objectives for the objectives bank, and a rudimentary computerized support system was developed to store, retrieve, and edit these objectives.

By the fall of 1979, several hundred basic instructional objectives had been collected. At that time it was decided to develop a prototype computerized IEP support system to demonstrate and refine the concept.

IEP SUPPORT SYSTEM PROTOTYPE (ORBIT-I)

An interdisciplinary project team was formed in the fall of 1979 consisting of a systems analyst, a programmer, and three special education teacher specialists who had been working on the objectives definition.

This team developed a set of requirements, assumptions and constraints used to define the system. These were:

1. The computerized IEP support system would store and retrieve objectives that were arranged in a simple hierarchy. This hierarchy would appear as follows:

```
GOAL AREA:
   TO DEMONSTRATE AN IMPROVED LEVEL OF INDEPENDENCE
   IN HOMEMAKING ACTIVITIES
--------------------------------------------------
    ANNUAL GOAL:
     WILL DEMONSTRATE SEWING AND NEEDLEWORK SKILLS
--------------------------------------------------
      SHORT TERM OBJECTIVES:
       WILL PREPARE NEEDLE AND THREAD FOR SEWING
--------------------------------------------------
       WILL MEND
```

2. Computer retrieval was planned originally to be by individual objective or by hierarchy - i.e. selection of items "low" in the hierarchical tree would also retrieve the higher items.
3. The computer system would be "user driven"; thus the special education staff would control both the data and the execution of reports directly via CRT terminals rather than through batch job submission.
4. The system would be implemented in schools on a purely voluntary basis.

The educational specialists quickly became enthusiastic about the data processing habit of using acronyms, and developed the acronym of ORBIT for the prototype system: Organized Resource Bank of IEP Text.

The technical approach chosen was dictated by the available MCPS computer software, the need to implement the system rapidly, and the nature of the prototype. These technical determinations were:

1. To implement the computer system using the BASIC language. The IBM VS BASIC package which ran as a foreground processor under Virtual Storage-Personal Computing (VSPC), an on line programming package, was used.

   These packages ran on the IBM 370-158 main frame computer and were accessible by using CRT terminals (IBM 327X).
2. Data input and job execution was to be via a CRT using VSPC and VS BASIC. IBM utilities and simple COBOL programs were to be used for reports (IEP's). Printing was to be done on the main IBM 158's printer and reduced by xerography.
3. The prototype computer system was to be kept as simple as possible.

Once these basic features had been determined, programming and further systems analysis then began on the newly designated ORBIT prototype system. The system was first demonstrated to the MCPS Special Education Parent Advisory Committee in November of 1979. The system became operational shortly thereafter, and is still in operation. Its operations are summarized below:

1. Teachers and staff participating in the project are provided with a printed "catalog" of the objectives in the bank. Order forms are also provided. Teachers refer to the "catalog" to select appropriate objectives which are suggested by the formal and informal evaluation results. The teacher indicates the selection of goal areas, annual goals, and short term objectives and prioritizes them based on the needs of the student.
2. The completed IEP order form is sent to the ORBIT project staff who key in the selection via CRT terminals. The ORBIT system then performs the selection programmatically, formats the objectives, and routes the selections to a printer.
3. The ability to prioritize at all levels of objectives (goal areas, annual goals, and short term objectives) was a feature added shortly after initiation of the prototype. Because of this addition, teachers can now better individualize and specify objectives for students. If behavior is the major area of concern, behavioral objectives can be selected and printed first even though reading and other subjects appear before behavior in the file.
4. The annual goals and short term objectives section of the IEP is copied and reduced to 8 1/2 X 11 size.
5. The IEP is then checked by the ORBIT staff, and routed to the teacher who made the request.

A schematic representation of the process is seen in Figure 1. The initial order form developed by the ORBIT staff listed goal areas, annual goals, and short term objectives. This order form required modification due to the increase in the number of short term objectives. It was no longer feasible to include a shortened version of the text for each short term objective. Since the order form would have been over ten pages long, it was necessary to implement a "fill in the blank" type of form, known as the "Computer Assisted IEP Order Form".

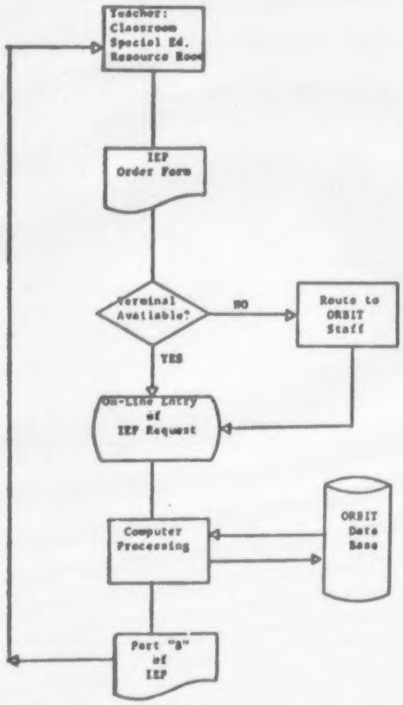


FIGURE 1
PROCESS FLOW: BUILDING AN IEP

Certain other improvements were made in the system over the past year. A feature was added to the IEP report which preprinted column headings for additional data to be inserted by the teacher, and a method for specifying objectives as present levels of performance was introduced. The most significant change was the growth of the objectives bank from 250 original objectives to over 2,000 objectives.

The current bank of objectives covers seven categories: specific content area (i.e. Mathematics, Reading, etc.), communication skills, sensory development, social-emotional skills, survival skills, vocational goals, and citizenship goals.

Logistically, the special education schools have used the computer-assisted IEP in the following manner. The teacher, after assessing the student's needs, develops an IEP objectives selection for the student and submits the order form to the ORBIT staff. The order form is processed, and an IEP is generated, reduced, copied, and returned to the teacher. The teacher then sends a copy of the IEP to the parent(s) of the student prior to an IEP conference. A note accompanies Part B of the IEP explaining that the document is a "draft copy" of the IEP being developed for the student. The parent(s) and teacher can modify the IEP as necessary at the IEP conference. If there were additions or deletions made at the IEP conference, a new order form is submitted to the ORBIT staff. To date the ORBIT staff has received no IEP's that have been resubmitted, but that option is still available.

## FURTHER DEVELOPMENTS

In the early spring of 1980, the ORBIT prototype system was demonstrated to staff of the Maryland State Department of Education (MSDE). Based on the demonstration and subsequent discussion, interest in expanding the ORBIT concept to include matching objectives with instructional materials, competency based tests, and instructional strategies was generated and MCPS was awarded an MSDE grant to support further ORBIT development including:

1. Establishing a system which could link a given instructional objective with (a) instructional materials suitable for teaching the objective, (b) instruments which could measure the achievement of the objective, such as compentency or criterion reference tests, and (c) instructional strategies (teaching methods) which could be used to teach the objective.

2. Providing teachers with instructional support guides which would contain the relationships between objectives, materials, tests, and strategies.

3. Initial development of the linked materials, tests, etc. in only one area — functional reading. This allowed an initial focused effort and also corresponded with the objectives of the Maryland Functional Reading Program, which will require passing a minimal competancy test for high school graduation beginning in 1982.

The grant supported part of the project from June to September of 1980, and resulted in the following products:

- The development of objectives, materials, tests, and strategies for reading was completed to the extent possible. An initial set of these materials was produced.

- The overall set of objectives was considerably expanded from 250 to over 2000 objectives.

- Progress was made in developing linked objectives, materials, tests, and strategies in other areas such as mathematics, and social-emotional skills.

- Design of an expanded computer support system was completed. Design and implementation of a data base structure was finished, and several of the on-line data entry CRT screens were finished. The first operational phase (objectives selection and IEP production) was implemented in April of 1981.

- Programming was completed to preprint as many data elements for sections of Part A of the MCPS IEP as could be extracted from the MCPS pupil database. The information was obtained from the student personnel record kept for all students and from special education census data kept for handicapped students only.

## Design Concepts and Requirements - Expanded IEP Support System (ORBIT II)

The improved computerized IEP support system was designed along the following lines:

1. It was determined to implement ORBIT II (the designation given to the expanded system) using more "permanent" software, and to integrate the application into the overall MCPS Data Base.

   The result of this decision was to develop the project using the IBM Display Management System (DMS) for the on-line data entry and inquiry functions, the Cullinane IDMS data base management system for data storage, COBOL for the load programs to the DBMS, and Cullinane's CULPRIT Report Writer for reports.

2. Based on experience with the prototype ORBIT I system, it was determined to continue the concept of a user controlled system. The system enabled the user to perform all data entry functions. IEP's were to be selected and printed at remote school sites. A feature to allow "browsing" through the data bank was seen as very desirable.

3. It was determined to print IEP's on a "letter" quality printer to avoid the supposed stigma of "computer" printing.

4. A top down design followed by a phased building block approach was developed to implement the following modules:

   A. Data Entry - On-Line DMS CRT displays for entering objectives, materials, tests, etc, and a change function for maintenance.

   B. Linking - A separate on-line function to allow the linking of objectives and materials, etc.

   C. Simple Selection - A feature to allow selection of specific predetermined objectives via the CRT and the printing of both the IEP and/or the Instructional Support Guide (selected objectives plus materials, tests and strategies).

   D. Generic Selection - A select feature based on generic selection keys - i.e. all reading objectives, etc.

   E. Browsing - An on-line module to allow specific or generic retrieval of objectives and further selection of the retrieved items as displayed on the screen.

   This feature would allow the precise selection of those items to be actually used for instruction. Thus, instead of all of the books that could be used to teach a math objective, the teacher could select for print only those items actually to be used.

## Conclusion

The concept of using the computer to store IEP objectives, and produce IEP objectives selected for a specific student is well established. Based on the simple and primitive ORBIT I system, users appear generally satisfied, and use is increasing. Parent reaction has been satisfactory.

Computer-assisted IEP's (Part B), were implemented in several settings in Montgomery County Public Schools and as of 12/31/80, computer-assisted IEP's have been generated for students, ages birth to 21 with the following handicaps: Speech Impaired, Mentally Retarded, Orthopedically Impaired, Other Health Impaired, Seriously Emotionally Disturbed, and Learning Disabled

Nearly eleven hundred computer-assisted IEP's for approximately eight hundred students have been generated in the period from 9/1/80 - 12/31/80. One hundred forty teachers, speech therapists, physical therapists, occupational therapists, and other specialists have participated in the program.

# Using A Microcomputer With A Large Data Base

Doug LaChance and Keith Stokka
2520 Broadway Drive
St. Paul, MN 55113

Abstract:
This paper will describe how a microcomputer is used as an intelligent terminal with a large computer system for educational administrative computing. The paper will identify the planning, development and implementation stages. It will include some facts about how data is entered, stored and later sent to a large computer for processing.

## INTRODUCTION

Administrative data processing for Minnesota school districts is provided by seven independent regional Management Information networks. The purpose of the networks is to provide equal access for the 435 Minnesota school districts in the areas of finance, personnel/payroll, and student services. Also, the need for uniform methods of gathering and reporting information to the State Department of Education and the state legislature helped bring rise to the current networks. Each school district within a region has access to the regional host computer through a variety of methods. One common method requires school districts to enter information onto input forms which are then mailed to the regional center for keypunching and input to the host computer. Another common method requires a school to purchase a terminal which is used directly with the host computer. This report will delineate an alternate method of processing financial transactions using a microcomputer in conjunction with a large host computer.

## PLANNING

In November, 1977, the Minnesota Educational Computing Consortium (MECC) established a Microcomputer Task Force which surveyed existing instructional usage of the small computers in Minnesota schools with the intent of developing a state contract for their purchase. The results of this effort are documented in a report titled MECC Microcomputer Task Force May, 1978.

A direct result of the task force study was a state contract in the fall of 1978 whereby Minnesota educational groups could purchase APPLE microcomputers through MECC. A few hundred purchases were expected but instead approximately 1,000 were purchased in the first two years. As of February, 1981, over 2,000 APPLES have been purchased through the contract.

MECC's first official interest in the use of microcomputers for administrative use came about by the preparation and publication of a report produced in May, 1979, titled A Feasibility Study of Administrative Uses of Microcomputers. The original purpose of the statewide microcomputer contract was for instructional use but an interest and to some extent, a demand, for use of microcomputers for administrative use was soon to follow. As noted in the May, 1979 report published by MECC.

> "The influx of a large number of APPLE II microcomputers into the school districts in Minnesota has led to an increased awareness among many school district administrators of the processing capabilities of computers, - particularly in small districts and in individual schools in large districts. Interest has been expressed in using APPLE II computers to handle some administrative activities.
>
> "As more Minnesota educators have gained some familiarity with microcomputers, opinions as to the potential value of computers for administrative tasks have been more

> prevalent and more extreme. Claims have been made at both ends of the continuum. Some say that microcomputers are capable of handling all of the administrative computing needs of small districts and schools, while others say that they are appropriate only for instructional computing needs. However, most educators who have an awareness of the microcomputers are simply asking for more information about the potential administrative applications that the machines might make feasible."

The May, 1979 feasibility study went on to identify the following potential administrative uses of a microcomputer:

### Potential Microcomputer Non-Instructional Applications

#### Student

1. Student Records (grades, locker numbers, courses, etc.)
2. Census (family)
3. Enrollment Projection
4. Attendance (daily - building)
5. Attendance (annual)
6. Athletic Eligibility List
7. Health Records
8. Mark Reporting
9. Student Scheduling Assistance (not computer scheduling)
10. Transportation (bus route development assistance information from census file)
11. Instructional Management (building level)
    a. CAM - Type
    b. Student Achievement
12. School Calendar (schedule of work days, holidays, teacher days, etc.)
13. Graduate Follow-up
14. Guidance Records
15. Test Scoring and Analysis

#### Personnel

1. Personnel Record (certification, seniority, etc.)
2. Salary Simulation
3. Paycheck Calculation
4. Assignment "System" (teaching assignments)
5. Payroll Related Reports (PERA, labor, sick leave, etc.)

#### Facilities

1. Facilities/Equipment Inventory
2. Energy Management (energy use accounting)
3. Facilities Utilization (percent of building utilization)
4. Maintenance (schedule of records)

#### Finance

1. General Accounting (budget, receipts, expenditures)
2. Accounts Receivable/Payable
3. Financial Forecasting
4. Lunch Program (lunch counts, inventory and reports)
5. Petty Cash Accounting
6. Vendor Reports and Purchase Orders
7. Certificates of deposit and Investments (interest rate, value at maturity)
8. General Ledger

#### General

1. Statistical Analysis (research activities)
2. Library Circulation
3. Media Reservations (equipment, scheduling, inventory)
4. Snow Removal Schedule
5. Project Planning and Budgeting
6. Activity Scheduling (extra curricular)
7. Word Processing (newsletters, etc.)
8. Mailing Lists/Labels (students, parents, staff, etc.)
9. Information Storage and Retrieval
10. Ad Hoc Reporting from Large Data Files

One of the recommendations made by the May, 1979 feasibility study states:

> "MECC should more thoroughly investigate the use of the APPLE II as a terminal device for the ESV-IS.
> One of the FY80 objectives is to examine the feasibility of using the APPLE II as a terminal to the regionally-based elementary-secondary-vocational-information systems (ESV-IS). The initial investigation indicates that there is potential for using the APPLE II as a terminal. As part of the investigation, a pilot demonstration should be established within a school district with fewer than a thousand students. The pilot should explore off-line data collection, editing, input, transmission, output, reception, and printing."

It was this recommendation that started the use of an APPLE as an input device for the statewide Finance System. (ESV-FIN)

## DEVELOPMENT

Once the development had begun, another study helped further enhance the concept. *Telecommunications and Microcomputers: A Study of the MECC Elementary and Secondary School Educational Computer Delivery System*, published in December, 1980, was funded by the Minnesota Legislature and produced by the MECC Special Projects Division. One of the objectives of this study was to review the current Management Information System network and develop an alternate microcomputer-based input system.

The follow excerpt from the *Telecommunication and Microcomputers* study indicated the philosophy at the time development began.

> "...the cost for telecommunciation is significant and increasing. Previous investigations conducted by MECC indicated that the APPLE II microcomputer could communicate with the MIS Burroughs computer systems. If much of the data entry work could be completed with the APPLE II

> prior to communicating with the host computer through telephone lines, possibly a less expensive telecommunications network could be advised. Also, the regional center directors believe that additional computer resources are necessary to handle the communications with the CRT's. Therefore, the use of APPLE II's could result in lower computer hardware expenditures at the regional center. An APPLE II - based system would allow for the creation and editing of MIS data batches without the aid of telecommunications lines. These batches would then be transmitted to the host MIS computer using telephone lines for only a brief time."

MECC-MIS was asked to do a segment of the study which was to pilot the concept of a microcomputer as an intelligent terminal to a large computer. The development began by writing programs for the APPLE that would allow the pilot district to enter transactions for both disbursements and vendor maintenance type batches. These two types were selected due to their high volume usage. The programs were written in a format to allow the APPLE to provide all the editing for the batches except for those which required database intervention.

In addition to the programs developed for the APPLE, modifications were made to some existing MECC software which allowed the APPLE to transmit and receive files from a large computer. Also, software was developed for the host computer (Burroughs) to accept the batch sent by the APPLE and attach the appropriate instructions to place the batch into the job queue for execution. The host computer also performed the tasks necessary to transmit the error report from the previously executed job back down to the APPLE.

With all pieces of software in place, the pilot district was ready to be selected. The pilot test was performed in the following manner.

1. The pilot district would continue processing their financial transactions in the usual manner of filling out input forms and sending them to the regional host for keypunching and execution.
2. MECC would copy a current version of the pilot district's database so that at the beginning of the test the pilot district would transact as usual against the regular database and also transact using the APPLE against the new copied database.
3. At month-end, processing reports would be extracted from both databases and compared. Identical line items and dollar figures within the reports would indicate a successful test.

Test runs for uploading (sending the batches to the host) and downloading (sending error reports from the host to the APPLE) were to be run at both 300 bits per second and 1200 bits per second using standard accoustical couplers.

The hardware configuration requirement for the pilot district included an APPLE with Applesoft and 48K memory, one monitor, two disk drives, and a 132 column tractor feed printer.

## PILOT TEST FINDINGS

Independent School District #511 at Adrian, Minnesota, was selected as the pilot district. The school had a student population of 696 and also had all the hardware required for the test. In addition, Adrian had been very active and innovative in their use of the APPLE in the past and were very anxious to try this new venture.

The pilot test began after a two-hour training session was given to Adrian on the use of the system. Minor problems encountered were quickly remedied.

At the end of the first month, reports were extracted from both databases. Examination of both reports showed them to be identical. The same examinations were made at the end of the second and third months with the same results.

Timings taken for uploading the batches to the host at 300 bits per second took an average of 3.7 seconds per transaction. A school district with the approximate student population of Adrian averages from 500-700 transactions per month. This would require 30-40 minutes of uploading time per month. Timings taken at 1200 bits per second were less than half that amount of time.

Timings were also taken for downloading reports to the APPLE . At 1200 bits per second, time taken to download a 30 page report averaged 55 seconds per page. With these timings, MECC is recommending that long reports still be printed on the high speed printer at the host computer, and that only short reports be downloaded to the APPLE.

Adrian found the system to be of benefit in many ways. Not only did it eliminate the mailing time that was necessary for the old method, but the actual time to enter the transactions was a lot less for the APPLE than for handwriting them on input forms. Secondly, many errors that were not found until the batch was run by the host computer previously were now being identified and corrected immediately before the batch was sent to the host computer. Thirdly, due to the simplicity and ease of operation, a clerical person at Adrian could easily be trained in the operation of entering transactions into the APPLE.

The Telecommunications Study points out some of the benefits and drawbacks for the use of the system from the regional host center's perspective.

> "If further testing continues to support the viability of APPLE II microcomputers as data building and forwarding devices for MIS, then several benefits may result from an adoption of this system by the regional centers. Possible benefits include:
>
> - keypunching services eliminated
> - reduction in editing problems requiring assistance by regional staff
> - reduction in load on computer system due to more error-free batches (pre ESV-FIN editing) and less CRT

communications

- increased district responsibility in the processing and timing of inputting their financial transactions

There are also some potential drawbacks to regional adoption of an APPLE II based input system. These drawbacks include:

- need for additional training of district staffs
- support for additional modes of input
- need for additional training for regional staff in the use of the APPLE II."

The chart shown in TABLE 1 taken from the Telecommunications Study contrasts three common modes of entry to the Finance System.

With the initial success of the pilot test, the following enhancements are currently being implemented.

1. Input screens with edits for all the commonly used transaction types.
2. The capability to print vendors checks in the district prior to sending the batch to the regional host.
3. The ability to execute any report available on the Finance System. (ESV-FIN)

We intend to pilot these enhancements for about three months. Assuming success, we would plan to expand the pilot to more school districts and address other questions, such as what data security problems arise, what problems are raised after more school districts use the same host computer with microcomputers.

In summary, we feel the distribution of work from the large/central processor to intelligent remote devices is an excellent option. While it was not our intent to discuss trends or predict technology changes, it appears distribution of work from the "Computer Center" will happen. What form this distribution takes is a decision for the users.

**TABLE 1 - DATA ENTRY METHOD**

| COMPONENTS AND FEATURES | BATCH FORMS REGIONAL KEYPUNCHING | DISTRICT ON-LINE CRT | DISTRICT APPLE II |
|---|---|---|---|
| School District Hardware Costs | none | $3100* | $3150 |
| Hardware Maintenance | none | $290-350/yr | $360/yr |
| Telecommunications Costs | none | high | low |
| Data Preparation and Input time | high | variable | low |
| Turn Around Time | high | low | low |
| Keypunch Services | $.05-.10/ transaction | none | none |
| Pre-ESV-FIN | no | no | yes |
| In-district check printing capability | no | no | yes |
| District control of ESV-FIN processing | low | high | high |
| District ability to enter transactions when Burroughs system is down | yes | no | yes |
| Ability to print short reports in district | no | no | yes |

*includes 2400 baud modem

# SOFTSWAP: THE EVALUATION AND DISSEMINATION OF NON-COMMERCIAL TEACHER CREATED SOFTWARE

Ann Lathrop
Library Coordinator
San Mateo County Office of Education

An educational software library for microcomputers has been established in the Microcomputer Center of the San Mateo County Office of Education in Redwood City, California. It is a joint project of the County Office and Computer-Using Educators (C.U.E.). The non-commercial public domain programs contributed to the Center by teachers and other educators have been evaluated, edited and organized into the SOFTSWAP, a free exchange of instructional software for classroom use. Approximately 200 programs are currently available in the SOFTSWAP for use with the five microcomputers in the Center: TRS-80, PET, Atari, Apple & Compucolor. Visitors to the Center may copy programs without charge and disks of programs are also available for mail-order sales.

The Microcomputer Center and SOFTSWAP in the San Mateo County Office of Education have grown from a dream into a reality in just over a year. The moving force behind the Center is a California based group of teachers and administrators, Computer-Using Educators (C.U.E.), who are dedicated to expanding the educational uses of microcomputers in schools. Administrators within the San Mateo County Office of Education, under the leadership of Superintendent William K. Jennings, worked cooperatively with members of C.U.E. to establish this resource center where educators can experiment with five different microcomputer systems, preview a wide variety of commercial software for use on these microcomputers and copy any of the approximately 200 public domain instructional programs that make up the SOFTSWAP.

## HISTORY

We receive many inquiries from educators in other areas who would like to replicate the Center that we have established. In our reply we stress the fact that none of the activities sponsored by the Microcomputer Center would be possible without the strong support of C.U.E. members. Computer-Using Educators was founded in 1978 by Dr. William (Sandy) Wagner, currently the Microcomputer Consultant for the Santa Clara County Office of Education in San Jose, CA. In less than three years the group has expanded from the small group of founders to a present membership of over 800, with members in 28 states and 8 foreign countries. Under the current president, Mrs. Bobby Goodson, C.U.E. is continuing to support a strong program of workshops, conferences and other meetings designed to expand the use of computer technology in education.

The SOFTSWAP originated as an exchange of educational software at the 1979 Annual Conference of the California Mathematics Council (Northern Section). This first venture was organized by Vince Contreras from San Jose State University. The programs contributed by participating teachers were deposited in the newly established Microcomputer Center in the spring of 1980. Teachers, administrators, students and other interested volunteers worked through the summer to evaluate, edit and revise these original programs. As new programs continue to flow into the Center they are routed to C.U.E. volunteers for evaluation and processing. Finished programs are placed onto SOFTSWAP Dissemination Disks which can be copied by anyone coming into the Center.

An increasing number of requests from outside the San Francisco Bay area encouraged us to begin a mail-order distribution service this spring. Our first catalog, which was published in the CUE NEWSLETTER, listed approximately 200 programs. We expect to receive more programs as word of the SOFTSWAP project spreads. These also will be evaluated and processed for dissemination.

## SOFTSWAP PROCEDURES

What makes the SOFTSWAP unique among the many users' group libraries available across the United States? The primary distinction is one of focus--

SOFTSWAP concentrates on the development of educational software. Most of the programs are short, stand-alone instructional units. Many are drill and practice exercises written for the elementary school or for remedial work at the secondary level. About one-third of the programs are math oriented. Most games have been excluded since these are generally available from many other sources. Only a few utilities, those of special interest to teachers, were on the first series of SOFTSWAP disks; these also are readily available from many user groups.

A second feature of the SOFTSWAP is the fact that every program is evaluated by at least two educators before it is added to a Dissemination Disk and made available for copying. Programs are edited for spelling errors, inaccurate or incomplete instructions, errors in factual content, programming errors and other problems. Our goal is to distribute only those programs which have some educational value and which are as free of errors as possible in both content and programming.

We have developed a very workable system for processing new educational programs donated to the SOFTSWAP. The entire procedure operates with part-time clerical staff, paid student assistants and volunteers from C.U.E.

Each donor is asked to complete a PRELIMINARY INVENTORY FORM (blue) that identifies and describes the program and a RELEASE FORM (green) that gives C.U.E. permission to distribute the program. The donor's form also asks whether the program is original, a modification of another program, or an adaptation from a magazine or book listing.*

Each donated program is first loaded onto a "Receiving Disk" where it will remain in its original contributed form as a permanent record. Filled"Receiving Disks"are placed into our permanent archives after all of the programs on that disk have been processed. One donated program at a time is loaded from the "Receiving Disk" onto a "Working Disk" and all subsequent versions of that program will be stored on the same disk until the program is finished and ready to be transferred onto the "Dissemination Disk." We have approximately 40 working disks, each with only one program, being evaluated at any given time. At least two educators will review the program on the "Working Disk" and make needed revisions. When they agree that it is finished the program will be removed from the "Working Disk" and added to the current "Dissemination Disk" in the SOFTSWAP. Another program from the "Receiving Disk" can then be loaded onto that "Working Disk" and the evaluation process will be repeated. All finished programs on the "Dissemination Disks" are available for copying by visitors to the Center.

We have two forms to aid in the evaluation of programs: CHECKLIST FOR MICROCOMPUTER PROGRAM REVISION (salmon) and PROGRAMMER REVISION SHEET (Yellow).* These forms list common errors and problem areas which the evaluators are to check. Detailed notes about needed revisions are made on the yellow form and the final reviewer checks to see that all requested corrections have been made. The finished program is described on a final INVENTORY FORM - SOFTSWAP (white) and then added to the SOFTSWAP catalog. Newly developed programs are also listed periodically in the CUE NEWSLETTER.

## EVALUATION OF SOFTWARE

The evaluation forms developed for use in the Microcomputer Center are not designed to produce critical reviews for publication. Our primary interest is in cleaning up a program, being sure that it runs well, eliminating spelling errors, factual errors, etc. The actual quality of the programs in the collection varies greatly, as does program length. Each disk contains from 12 to 30 programs, all for one system, but for various subjects and grade levels.

Each program is checked for accuracy of content, clear instructions, trapping of input errors, etc. Certain problems which frequently appear in contributed software could be reduced or eliminated if donors would ask themselves the following questions:

1) Does the program run with relatively few problems? (Unfortunately, we receive many programs which do not load/run at all.)
2) Does the program trap input errors? Does it crash if the user inputs numeric response when the computer is expecting alpha, or vice versa? What happens if the user hits an unwanted ENTER/RETURN? When there is an input error, is the computer response clear and simple? "Input error, redo from start" is not helpful to the novice.
3) Are all words spelled correctly?
4) Are the instructions and other information easy to read on the screen? It is best to avoid crowding and to clear the screen at appropriate intervals. Instructions should be written at a reading level appropriate for the intended user.
5) Is the POSITIVE reinforcement for correct responses more interesting than the negative reinforcement for errors? Graphics in most HANGMAN games, for example, are usually much more exciting when the user loses and is hanged. Students will often deliberately miss a problem or word in order to see a rocket explode, a boat sink, etc. Try to provide strong encouragement for doing the task correctly and yet avoid over-praising.
6) Have you included REM statements as needed? They make it easier to use a program effectively. It is especially helpful to include statements telling the teacher how to modify word lists, problems or other entries.
7) Do you make negative remarks to or about the student? Sarcastic statements can be discouraging. We remove words like "Boy was that dumb," "Stupid," "Dummy," etc.

---

*forms available from the author at the San Mateo County Office of Education, 333 Main St., Redwood City, California 94063

8) Do you provide hints and/or eventually give the user the correct answer? We usually allow only 3 or 4 incorrect responses before providing either a hint or the correct response, often with some instruction or explanation. Exceptions are programs which are tests or timed drills for accuracy.

Two other factors are considered when evaluating programs for the SOFTSWAP. We try to be sure that the program submitted is truly an original creation of the author. If it is an adaptation from another program or from a book or journal we want to be aware of that fact. We will then request the author's permission to include the program in our collection. If we discover that a SOFTSWAP program is actually an infringement on someone else's copyright we remove it from the dissemination disk.

Finally, the most exciting programs which we receive are the creative ones that do more than just put workbook pages and drill problems onto the computer screen. Software is improving as educators become more sophisticated programmers. Newer programs are often more challenging to the student user and make far better use of the unique capabilities of the computer. We are especially pleased when we receive donations of this type of program and take special pride in being able to make them available to other teachers through the SOFTSWAP.

## PREVIEW OF COMMERCIAL SOFTWARE

The third function of the Microcomputer Center is to serve as a display area for copyrighted commercial educational programs. Many software firms have provided sample copies of their instructional materials for preview by visitors to the Center. We have also been able to purchase some exemplary programs with funds from the San Mateo County Office of Education. Our goal is to make available selected software for each of the five microcomputer systems in the Center in order to demonstrate the special features of the system. We also want to give the potential user some idea of the range and quality of materials available in various subject areas at both the elementary and secondary levels.

## THE MICROCOMPUTER CENTER

A unique feature of the Center's development has been the close and continuing cooperation between Computer-Using Educators and the San Mateo County Office of Education. Visitors to the Center frequently ask about our funding. To date we have not received any outside grants or other financial aid. The Center is supported in part by the County Office as a service to the educators in our county and throughout the San Francisco Bay area. Hundreds of hours of volunteer time are donated by C.U.E. members as they participate in the educational activities and demonstrations scheduled at the Center. C.U.E. members also contribute the expertise needed to evaluate and edit the instructional programs in the SOFTSWAP. County Office staff, including the author, provide the organization necessary to maintain the SOFTSWAP and other projects.

Manufacturers of microcomputer hardware have also been generous in their support of the Center. Tandy Radio Shack and Commodore PET provided the first microcomputer systems and thus initiated the support that has made possible all subsequent development. Each firm set up a complete system with microcomputer, dual-disk drives, printer and cassette-recorder--all on long-term loan to the Microcomputer Center. A library of commercial programs loaned to the Center by each firm includes educational software and word processing systems.

Other firms providing equipment for the Center include Atari, Apple and Compucolor. All of the systems are on long-term loan; the Center has not purchased any hardware. C.U.E. members were of great assistance in contacting the various firms and securing their support and cooperation.

All of the manufacturers have been helpful in maintaining the equipment in good condition, despite the heavy use in the Center. They also provide new models as they are developed. This strong commitment on the part of the manufacturers has been a crucial factor in our success.

## STEERING COMMITTEES

The development of the Microcomputer Center has been characterized by cooperative interaction of County Office staff from several divisions within the office. The Microcomputer Center is located in the Educational Resources Center Library, under the direction of Dr. Curtis May. The SOFTSWAP is coordinated by Ann Lathrop, author of this paper. Dr. James Fee, Director of Media Services, offers assistance with hardware applications and demonstrations. Curriculum development programs and in-service training are the responsibility of LeRoy Finkel, computer consultant, and Walter Smithey, math-science consultant, working under the direction of Tom Quinn. Dr. Al Grossman, Administrator of the Office of Information Services, has developed an extensive program that will interface computers in the County Office to the microcomputers in the Center and in schools throughout the County.

Utilization of this team approach assured a broad base of support and has made the Center responsive to the needs of many different groups within San Mateo County. The entire project is under the administration of Dr. Don Halverson, Assistant Superintendent for Educational Support and Planning.

Four members of Computer-Using Educators served on the Steering Committee for the Center and for the SOFTSWAP. Jim Love (PET), Brian Sakai (TRS-80), Dave Stone and Bob Enenstein (Apple) contributed the time and effort necessary to guide the Center through its first year and to create the SOFTSWAP.

## SUMMARY

We have achieved our goals of providing equipment for demonstration, commercial software for evaluation, and public domain programs which can be copied by teachers for classroom use. The working partnership between C.U.E. and the San Mateo County Office of Education has created a Center which we feel has a significant role in the developing uses of microcomputers in education.

# LEARNING BY COMPUTER PRACTICE

E. Jay Lindsay
Time Share Corporation
Hanover, New Hampshire

Educators have largely ignored in their research practice as a way of learning, and have explored only very few innovative ways of providing practice. Yet, practice is often recommended as the best remedy for some learning deficiencies. As important as this learning technique is, it seems prudent to investigate more fully its potential. As discussed, the computer offers one means of providing practice that is effective, stimulating and fun for students. An example is presented and reasons why computer practice is effective are examined.

It is a curious matter that educators have largely ignored practice as a teaching technique in their research and classrooms. In one extensive study of teaching, for example, no mention is made of practice as a method to increase student learning, notwithstanding the large variety of teaching and learning techniques discussed (Duncan, 1974). Many similar examples could be cited.

Reasons have been suggested why this might be so: The approach seems "unprofessional", it is not as stimulating for teachers as other teaching techniques, it requires of teachers an enormous amount of time and energy, and according to teachers it is boring for students (Williams, 1979). The intent of this discussion is to dispel such negative notions. Practice is needed. It can be fun and highly stimulating for students. And it can be rewarding for teachers. The key to this kind of practice is the computer.

Because of the nature of the computer, computer-based learning systems offer a kind of practice that is very effective. Many aspects of instruction associated with high achievement are easily incorporated within computer-based learning programs. The result is that students achieve substantial learning gains when their normal classroom istruction is supplemented by computer practice. Furthermore, the tedium and time required of teachers to provide individualized practice to several students is essentially eliminated by use of computers.

In 1977, Time Share Corporation (TSC)[1] began development on the Dolphin Educational System,[2] a set of computer-based learning programs in reading, language arts, mathematics, and a newly developed program in mathematics problem solving. These programs provide a highly motivating form of personalized practice.

---

[1] Time Share Corporation, a Houghton Mifflin subsidiary, produces computerized educational programs in career guidance, basic skills instruction, and classroom management.

[2] The Dolphin System is also known as STRIDE. The name Dolphin Educational System has been adopted since the recent introduction of TSC's 8-terminal micro computer, called the Dolphin Micro.

The curriculum of all programs combined consists of over 35,000 practice and test items, providing 600 hours of instruction.[3] Much of the curriculum is based on proven Houghton Mifflin material.

When used in conjunction with normal classroom instruction, Dolphin programs have produced significant and consistent gains over what would be expected from classroom instruction alone. Studies indicate that students who use the Dolphin system gain, with only a few months instruction, from a half to more than a full year's learning over what would be achieved normally.

I will only discuss in this report Dolphin Reading and Dolphin Language Arts. These two programs are identical in concept, structure, and grade level coverage (three through eight),[4] but of course, differ in their curricula.

## The Dolphin Rationale

Central to the Dolphin system are teachers. Dolphin programs are designed around the concept that teachers know their students best, and as such, should have complete flexibility in determining each student's individualized curriculum. Teachers much prefer to have this control (Swinton, 1978) and, in fact, feel a loss of esteem and satisfaction when they are not allowed to make such decisions. Our studies confirm the obvious: Dolphin instruction is much more effective when it matches student need and is correlated with classroom instruction, a task best accomplished by teachers.

## The Dolphin Educational System

The foundation to all Dolphin programs is high quality, comprehensive curricula, divided into separate independent learning units, called skills. In Dolphin Reading, for example, "Details", "Main Idea", and "Cause/Effect" are three skills in reading comprehension. The reading curriculum consists of 418 word attack, vocabulary, comprehension, and study skills. The language arts program has 449 skills in grammar, usage, and mechanics.

The Dolphin student. Immediately after students sign on the computer, they begin instruction in a skill assigned to them by their teacher. They "enter" a skill at one of the three skill stages -- Pre-test, Practice, and Post test -- depending where they finished in the last instructional session.

Pre- and post tests are five item criterion reference tests taken directly from Houghton Mifflin's nationally validated test battery, the Individual Pupil Monitoring System. To insure high reliability Dolphin test items are identical to the "paper" test items, and as such are multiple choice format, with no corrective feedback being provided. An example of a sixth grade level pre-test item for the Cause/Effect reading comprehension skill is shown in Figure 1. Both pre- and post tests consist of five such items presented in succession.

Contingent on pre-test performance students are either given a set of practice items or they proceed to the next skill. If students answer four out of five pre-test items correctly they begin a new skill, otherwise they receive practice.

The practice stage of a skill begins with a "Note" to students about a key aspect of the skill. The purpose of the "Note" is to present a definition, concept, or other information useful in answering practice exercises and in understanding the skill. As an example, the "Note" for the language arts skill on predicate adjectives is shown in Figure 2.

Unlike pre- and post test items, practice item formats are of a large variety -- the typical true/false, multiple choice, and fill in the blank variety, plus a large number of nontraditional types. Item formats are varied to increase student interest and to allow students to view skills in different ways, the rationale being that this will produce more learning. Typically, three to five different formats are used in any one skill.

Practice items, shown in Figure 3, illustrate three formats. As indicated, questions are preceeded by directions, the skill number, and a brief description of the skill. The first two examples depict exercises of word attack and vocabulary skills. The third example, shows three successive exercises from the skill on identifying the correct form of irregular verbs.

---

[3] These numbers represent the amount of instruction students would receive if they were presented all materials in every grade level.

[4] Grades 3,7, and 8 for the Language Arts program and grades 7 and 8 for the Reading program are new this year.

### Figure 1

READING
SKILL #644 PRETEST - Cause/Effect

DIRECTIONS:

Read each numbered group of sentences. Decide the order in which the sentences should be written in order to tell a story. Then find the correct answer.

1. Walking along the boardwalk, we noticed a large crowd gathered around a store window.
2. There we saw a man in a tall white hat making pancakes on a long griddle right in the window.
3. Naturally, we hurried over to see what was going on in the window.
4. They looked so golden and good we went right in and had some.

3. Which is the correct order of the sentences?

A) 1,2,3,4    B) 1,2,4,3

C) 1,3,2,4    D) 1,4,3,2

### Figure 2

LANGUAGE ARTS
SKILL #612 PRACTICE - Predicate Adjective

Just a note...

Sometimes the word that follows a linking verb describes the subject. Look at this sentence and notice the underlined word.

Jon is clever.

The word "clever" is an adjective describing the subject "Jon". We call such an adjective a PRECIATE ADJECTIVE.

As is always the case, students are presented a sample exercise when a new format is introduced. In this skill, practice questions one through seven are of one format, while eight through fifteen are of another. For the sample, the students are asked to type the past form of the verb come. Their response, CAME, is shown in capital letters. Subsequent items are presented one at a time with all answered items remaining on the screen for review by students. The 10th item has yet to be answered.

Each practice session is ended with a "Reminder", summarizing for students what they have just learned. The "Reminder" is a review of the "Note" presented at the beginning of practice.

### Figure 3

READING
SKILL #607 PRACTICE - Endings, Plurals

DIRECTIONS:

Read each sentence and notice the underlined word. Type the singular form of the underlined word.

6. Our garden seems to grow tomatoes best.

READING
SKILL #632 PRACTICE - Synonynms/antonyms

DIRECTIONS:

Read each sentence and notice the underlined word. Type the word from the list with either the same or the opposite meaning to go in the blank and complete the sentence.

| | | | |
|---|---|---|---|
| dreadful | bothering | curious | generous |
| polite | engine | seldom | mend |
| tasty | experienced | new | wise |

3. My mother is always courteous behind the wheel. She says, "A (        ) driver is a safe driver."

LANGUAGE ARTS
SKILL #631 PRACTICE - Irregular Verbs

DIRECTIONS:

Type the missing verb tense.

| | Present | Past | Past participle |
|---|---|---|---|
| Sample: | come | CAME | had come |
| 8. | go | WENT | had gone |
| 9. | know | knew | had KNOWN |
| 10. | | took | had taken |

A simple but important aspect of the Dolphin System is that it compliments and reinforces students when they have correctly answered a practice question and corrects them for wrong answers. A few examples of comments are:

Good!
Correct, you are doing fine.
Right, Teddy!
Jackpot! You got them all right.
The right answer is WINDOW.
Please try again.
We'd like to give you more help, Freddy.

It is surprising how motivating these comments are to students. They are delighted that the computer "knows" their name. And they seem to be just as responsive to the comments at the end of the school year as they were at the beginning.

When students have finished the practice section of a skill, they proceed to the post test. Post tests are equivalent to pre-tests. They have the same format, reliability, and similar content. As with pre-tests, four out of five correct is the criterion for pass, in which case students begin instruction on the next skill.

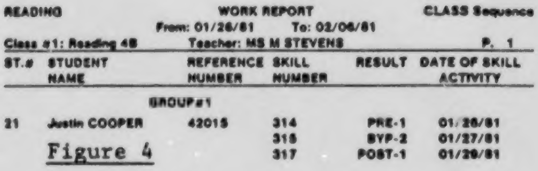
If students fail the post test or answer the first five practice items incorrectly, the Dolphin program automatically refers them to off-line assistance. What action students take at the time of a referral can be more or less automatic. Teachers can "program" into the Dolphin System a set of referrals for each skill that will be presented to students when they have unsuccessfully completed a skill. A referral usually directs students to their teacher, to specific pages in a reading text, or specially prepared material.

The Dolphin teacher. An important aspect of the Dolphin program is the management system that allows teachers to assign skills and to monitor student progress. Skills can be assigned in two ways. One way is by the Survey where teachers can have presented to their students a diagnostic test on any set of skills. One to five post test items are randomly sampled for each of these skills and included on the test. The number of post test items sampled is the same for each skill, and is specified by teachers. After taking the Survey, students are automatically given instruction on only those skills in which they did not answer correctly all sampled post test items.

Alternatively, teachers can assign skills such that students will receive instruction in one skill (i.e., progress through the three skill stages) before continuing on with the next skill.

In both ways, individual or groups of skills can be assigned to single students, to an entire class, or to groups of students within a class. Skill assignments can be done easily and quickly by editing commands that allow adding, deleting, and inserting skills.

To aid teachers in making skill assignments, the Dolphin programs provide several reports that summarize class enrollments, skill assignments made thus far, survey results, and student progress. A very useful and much used report is the Work Report shown in Figure 4. Although the example only shows one student, in practice all students within a class are listed. For each student the Work Report lists the skill (by number), the outcome or result of that skill, and the date of the result. The example illustrates that "Justin Cooper" passed skill #314 the first time he took the pre-test (PRE-1); bypassed skill #315 after two failures (BY-2), allowing him to continue with the next skill; and passed skill #317 on the first post test (POST-1).



READING WORK REPORT CLASS Sequence
From: 01/26/81 To: 02/06/81
Class #1: Reading 4B Teacher: MS M STEVENS P. 1

| ST.# | STUDENT NAME | REFERENCE NUMBER | SKILL NUMBER | RESULT | DATE OF SKILL ACTIVITY |
|---|---|---|---|---|---|
| | GROUP#1 | | | | |
| 21 | Justin COOPER | 42015 | 314 | PRE-1 | 01/26/81 |
| | | | 315 | BYP-2 | 01/27/81 |
| | | | 317 | POST-1 | 01/29/81 |

Figure 4

The final way teachers interact with the Dolphin program, as discussed earlier, is through the referral system. This feature is consistent with our philosophy of teacher control and has been used very successfully by teachers. Developing referrals is an effective way for teachers to become thoroughly familiar with the curriculum, an ingredient necessary for insuring optimum results.

## Research Results

The Dolphin programs have been well received by both teachers and administrators. Reasons indicated for their satisfaction are: The capability to designate what should be taught to each student, a skill-oriented curriculum, the management system, and student motivation and achievement. With regards to the latter point, Figure 5 summarizes evaluation results of the reading and language arts programs. These data are from two experiments in which students were randomly assigned to receive Dolphin instruction or to not receive it. Students in these two groups were matched on initial ability, as measured by the California Achievement Test (CAT) for the reading study and the SRA achievement test for the language arts study. In both experiments students received about three months instruction in 20 minute sessions, three to four times a week. The results are as indicated: Students using the Dolphin gained an additional four months in "reading comprehension" and additional year and two months in "language usage."

More detailed analyses emphasize that for best results the Dolphin System should be used as an adjunct to regular classroom instruction and should be closely correlated with the daily lessons given by teachers (Lindsay, 1981).

Figure 5

| | N | Pre | Post | Gain |
|---|---|---|---|---|
| Dolphin students | 95 | 6.1 | 7.3 | 1.2 |
| Control students | 95 | 6.1 | 6.9 | .8 |

Reading Comprehension (CAT)

| | N | Pre | Post | Gain |
|---|---|---|---|---|
| Dolphin students | 37 | 7.0 | 8.8 | 1.8 |
| Control students | 37 | 7.0 | 7.6 | .6 |

Language Usage (SRA)

## Why is the Dolphin Effective?

The type of instruction provided by the Dolphin System is simple in concept. Necessarily, all details and future developments have not been described. But in any event, the key emphasis of the Dolphin programs is mainly to provide practice. This is true even for the new mathematics problem solving program which includes tutorials, special exercise formats, and other innovations. The basic structure is the same: Students first receive in-class instruction on a skill or concept and then practice on what they have learned by working problems and answering questions presented by the computer. The fact is that this kind of instruction, although simple, is effective. Why?

There are obvious factors that effect learning: The curriculum, management system, individualized instruction, the monitoring of student progress, and continuous positive reinforcement. One factor, however, deserves special attention, because it is the focal point to almost every facet of a computer-based learning program.

Carroll (1963) proposed a model of school learning in which time was the central variable. In his model he made the important distinction between elapsed time and time the learner actually spends on the act of learning. Recent studies have shown that this latter time -- academic engaged time -- is highly correlated with achievement. This result of course is obvious. It simply says that the more one focuses on what is being learned, the more one will learn. This concept is perhaps no where more important than in computer-based learning programs like the Dolphin. Nearly every feature of the Dolphin tends to maximize the amount of time students spend actually learning. The interactive instructional dialogue requires and motivates students to concentrate on what is being taught. Reinforcement and dynamic cueing foster greater attentiveness. Teacher control minimizes the possibility of students working on inappropriate and irrelevant material. And the mastery learning structure,[5] through its affect on other variables -- self confidence, preparation, rate of instruction -- tends to increase the amount of time students spend learning as they progress through the curriculum (Bloom, 1976).

Although, TSC has not objectively studied how Dolphin instruction affects academic engaged time, it is obvious from observations that students are intensely involved in their use of the Dolphin. Rarely, if ever, do students look anywhere other than at the terminal screen or engage in any activity other than using the Dolphin. It seems, therefore, that a student's learning time is nearly all spent on meaningful learning activities. Our conclusion is that, in light of recent research, this facet of the Dolphin -- that it seems to maximize academic engaged time -- is a very significant reason why it improves learning.

## Conclusion

There are many reasons why practice provided by computer is effective. More importantly, within the basic model -- Pre-test, Practice, Post test -- there are highly interesting and attractive uses of the computer. Such possibilities have only begun to be explored.

Practice is needed. It is effective. It is highly stimulating and fun for students and rewarding for teachers. And to return to the original point: The key to this kind of practice is in the computer.

---

[5] Mastery learning refers to self-paced instruction in which the learner must achieve a specified level of mastery over a skill before continuing on with the next skill.

# BIBLIOGRAPHY

Bloom, B.S. Human Characteristics and School Learning. New York: McGraw-Hill, 1976.

Carrol, J.B. A model of school learning. Teachers College Record, 1963, 64, 723-733.

Dunkin, J.D. & Biddle, B.J. The Study of Teaching. Holt, Rinehart and Winston, 1974.

Swinton, S.S., Amarel, M. & Morgan, J.A. The PLATO Elementary Demonstration Educational Outcome Evaluation, Final Report, ETS, Nov. 1978.

Williams, J. Reading Instruction Today, American Psychologist, 1979, 34, 917-922.

# MICROCOMPUTERS IN TEACHING HANDICAPPED STUDENTS

by

Duane Loewen
MECC Instructional Coordinator
Moorhead State University
Moorhead, MN 56560

Ten years ago the major constraints of using computers with the handicapped were the unavailability of computers, expense of the hardware, and lack of software. Essentially what was present were innovative instructional ideas using computers but were limited to a "laboratory" situation because of the previously mentioned constraints.

The microcomputer has brought the giant unwieldy computer capabilities from the temperature - humidity controlled rooms in the universities to the small local school classroom. What cost $20,000 yesterday cost $1,500 or less today. Additionally, today's microcomputer offers capabilities not previously available.

Present computer technology offers a viable option for implementing IEP goals, objectives and activities which can focus on specific individualized needs of handicapped students. The use of microcomputers is an ideal tool for assisting in mainstreaming special students with other students who are running their own individualized computer programs. The implementation of these ideas can be readily done by using existing school microcomputers. Simple modifications of hardware and/or software can be readily performed by local school media personnel. Following is a discussion of ways computers can facilitate meeting individual educational goals for representative categories of students.

## Mental Retardation

The uniqueness in intellectual growth and performance demands a highly individualized approach, especially when concepts being taught are cognitive in nature. A breakthrough in the attempt to individualize learning experiences came with the implementation of programmed instruction techniques.

The characteristics of programmed instructions correlate very closely with individualized educational plans including:

1) Stating precise educational objectives in behavioral terms.
2) Presenting material in consistent, logically sequenced order in minute steps designed specifically for meeting stated goals and objectives.
3) Providing consistent, immediate and ongoing feedback designed to meet students' specific preferred forms of reinforcement.
4) Allowing the students to proceed at their own learning rate using a program which is tailored to their specific physical and learning characteristics.

A major constraint such as those presented by computer assisted instruction has been the student's need to read printed texts on the computer screen. A major breakthrough - low cost speech synthesizers chips - allows for spoken lessons to be presented along with the text which is printed on the monitor.

The microprocessors also provide branching capabilities which allow students to receive programming that addresses the specific needs of each student.

Along with the capability of extensive branching is immediate feedback relative to the appropriateness of the student's response. Simplified programming methods also allow the teacher who has a minimum of programming skills to add, delete or modify existing programs. The capability for retrieval of an enormous variety of curricular material within minutes presently exists.

Computers can facilitiate the reaching of specific educational goals of students:

1) Lessons are sequenced according to student's responses.

2) Immediate and individualized feedback and reinforcement are presented.

3) Slide projectors, tape recorders, lights and other equipment can be controlled by student's responses.

4) Various reinforcement techniques such as tokens, bells, music are easily programmed.

In addition to the previously mentioned attributes, the computer patiently presents virtually any concept over and over until it is mastered without passing judgment on the student's performance. Other advantages concerning the use of computers in the instruction of retarded students include:

1) The systems are individualized so that students can use them in the presence of their peers without fear of embarrassment.

2) Data on a student's performance can be kept and readily analyzed to present students with instantly updated and revised program material.

3) Data on a student's program is instantly stored and processed providing for accurate base line data and accountability of student performance.

Computer managed instruction is another dimension of potential computer use in the education of the mentally handicapped. In this mode the classroom teacher continues to present lessons; however, the actual course of the instruction is individualized by means of providing information regarding students' responses to lessons. The output consists of information about lessons and correlated tasks which may be incorporated to enable the child to reach a specific instructional objective.

Visual Defects

Visual defects may range from mild sight defects to total blindness. Many adaptations in the output of computer systems may be incorporated to make a computer system accessible to one with visual impairment. The most obvious means of making computerized instruction available to the visually handicapped is to incorporate speech capability. This technology is presently available and will become more readily available as technology advances. Earphones can be worn to insure complete privacy and to prevent interferring with other class members.

A simpler system for those with mild vision defects is the incorporation of high resolution graphics and the use of highly readable and visible color contrast such as black and yellow combinations. Written text may easily be typed in and placed on the monitor and then stored on diskette. Text may then be placed on video tape cartridges for viewing.

Motor Involvement

Some handicapped individuals may have motor involvement and poor coordination. A good example is an individual who has cerebral palsy. It would be almost impossible for such an individual to correctly input information into a computer system. However, a simple sign on procedure can be executed by passing an identification card through a reader. The card could contain just the information needed to sign on and in some cases the actual program desired could be encoded on the card.

Other means of facilitating input could be incorporated by enlarging the keyboard thus increasing the size of the target area and making accuracy less critical. Also, one or two touch panels which are well separated can be used to input a simple yes or no, true or false response. This format limits the input to two responses but this is adequate for drill and practice material and existing programs which have been modified to accept this type of input.

A program can easily be written to enable those with severe lack of coordination to operate a computer. This system may incorporate only a few well separated keys which can be pressed according to a specific code to provide input accessibility to all characters found on a standard computer.

The technology to make voice input is becoming a viable means of inputting commands. A vocabulary of two or three words which were "understood" by a computer would enable a totally paralyzed individual to run a modified program.

Severe and Profound

Existing computer technology allows for using computers as a means of eliciting and presenting responses from even the most severely handicapped students. Words, symbols, such as Bliss Symbols, graphics and phrases can be accessed and presented visually or spoken. By modifying input systems using touch switches or photo cells, the most minute student movements may be detected and reinforced.

The most pressing need in the educating of severe and profound students is to enable the student to make detectable and consistent responses. These responses provide a reliable source of feedback to be used as a basis of future programming and interaction with the student.

Summary

The computer can be used in a very effective manner in the education of handicapped students. When the computer is used to provide programmed instruction to the severely handicapped, adaptive devices are necessary. Using this special technology allows the student to function much more independently than could ever be possible without the computer. There is very high potential for providing a highly effective means of instruction. The extensive capabilities of the computer for branching and record keeping make it highly efficient for individualizing instruction. The population under consideration desperately needs the assistance.

Questions will be raised about the cost and feasibility of implementing computer assisted education for handicapped students. If computerized programs for the handicapped are aimed primarily at making a handicapped individual a potential wage earner and contributing member of society, the time and cost for new and innovative methods can be justified. The low cost of microcomputers allow for very cost-effective application of technology to the education of the handicapped.

Emphasis on meeting the needs of the handicapped has led to innovations in educational technology. The strongest justification for extensive expenditure of time, effort, and money can be based on the fact that a computer can react to a student at his own level for an infinite amount of time with an infinite amount of patience. This single attribute greatly increases the chances of a handicapped individual attaining his full educational potential.

# ADMINISTRATIVE USE FOR THE THIMBLE:
# A MICRO-COMPUTER ATTENDANCE SYSTEM

Paul Lorton, Jr. & Eugene J. Muscat

University of San Francisco
San Francisco, CA 94117

Attendance is becoming increasingly important. The attendance based support from the State in California has risen markedly over the past few years. In addition more schools have access to micro-computers which can be used in the monitoring of attendance at the school site. The key problems in using these systems for attendance are providing for flexibility and ease of operation at the school site and insuring compatibility with a variety of systems at the central office site.

Working from a base of ten years experience with a variety of computerized attendance systems, a micro-computer based system has been developed which allows easy, flexible data entry and accurate, timely information for the school site administration. This system has been developed for a moderately low cost micro-computer configuration which is also frequently found in schools.

As a model, the system allows computers acquired for instructional purposes to be used by school site administrators with little or no impact on the instructional program.

## INTRODUCTION

Micro-computers are appearing in schools faster than they are being used. With a little extra money in a budget, a micro-computer can be purchased and a use devised later. They will certainly be used. Classroom teachers in math, English, and other subject areas have a range of applications which increases daily. However, the school administrators are also looking at these small wonders with more than casual curiosity. They are asking what important administrative functions can be accomplished with the small but powerful equipment.

Attendance is an important function. More and more, the responsibility for student performance in school is being placed on the school. The first area where this is often manifest is in making sure the student is in school. School site administrators have felt increasing pressure and an increasing need to know the attendance behavior of their students.

Attendance, a usual base for funding school operations, has historically been a function accomplished by collecting information at the school site and passing it to the central office with little or no processing by the school site administration. Reports are then generated at the central office level and passed on to other agencies such as the State for Average Daily Attendance (ADA) funding. This approach made little but the most perfunctory use of the information collected by those at the school site.

## CINCH

In the 1960's and early 1970's additional effort was made to use attendance information for other than 'ADA' purposes. One project, which we reported at an earlier AEDS conference (Lorton & Muscat, 1976) used a small timesharing system located at a secondary school site to collect attendance information in order to provide timely reports to the home about students' attendance behavior.

The main purpose of the CINCH project was to produce reports for the home. The information was collected on mark sense cards. This was necessary in order to collect information detailed enough to be useful with minimum impact on the normal functions of the classroom teacher and the school administrators. Collection of the attendance information in machine readable form meant that the data could be processed from the classroom into the machine with little handling. The basic design was dictated by the need to produce timely information at the lowest possible human and monetary cost.

The system was quite effective and well enough received to inspire the central office of the school district to include many of the features tested by CINCH in a subsequent daily attendance system implemented on a large IBM system. The large system used light pens for data entry from attendance strips produced by the computer and marked by the classroom teacher.

Rather than simply collecting information to be passed onto other agencies, attendance data was used to inform personnel at the school site on the attendance patterns of their students. Students who were cutting classes could be identified to their deans, counselors and home. Since, in the CINCH model, the school site personnel initiated and were responsible for the information communicated to the home, the coordination of home and school efforts to improve attendance was enhanced.

## CURRENT NEED

School site personnel have the opportunity for the greatest impact on students. The experience with the school site attendance system and with the multi-school system implemented by the central office was important in the development of a set of necessary features for an attendance system. The features include:

### Detailed Collection

While the impact on the classroom teacher must be kept to a tolerable level, as soon as the school personnel realize the potential for a locally maintained attendance system, they want reports which can only be developed from detailed, daily and class by class information. The system must start from a design which allows period by period daily recording of absent and tardy behavior.

### Automatic Handling

The effort to collect detailed information while keeping the costs low means minimizing the personnel involved in handling the data. One of the current principles in devising management information systems warns that the data must be entered into the system as close to the source as possible. From the source of the information to the production of reports, the processing should be under machine control.

### Flexible Reporting

As school personnel realize what can be done, they want to make use of the information. Listing all the information contained in the system for every requestor is never effective from either a cost or utility standpoint. Reports must be produced to meet specific needs and to respond to specific requests. This means that the system for reporting information from the collected data must be flexible so that it can meet the changing requirements placed on it as the using audience becomes more sophisticated.

Recent studies in the San Francisco Unified School District have indicated what can be accomplished with this sort of effort. A process has been developed (see Harrington, Scourkas & Lorton, 1981) which provides for extensive analytic analysis of period by period attendance information. The cutting behavior of students in four San Francisco High Schools was analyzed in detail and various patterns were determined which will provide useful information to the school administrators at each of the secondary schools.

In order for cutting behavior to be studied in detail, the school site administrators need the information at their finger tips. This can be accomplished in several ways. Some school districts do have the central data processing capacity to place terminals from a central system at the school site. Others do not. Even in those districts with large central computing facilities, the responsiveness of the central data processing staff to school site needs is not always immediate or even timely.

We have argued in the past that instructional use of a computing system could be accomplished at little or no cost (see Lorton & Muscat, 1974 and Muscat & Lorton, 1978). By performing certain administrative tasks on the computing system in parallel with student use and/or at times when the system was not needed for instruction, the support of the system could be underwritten by the administrative uses. With the advent of micro-computers, affordable computing was available to the classroom teacher without having to worry about supporting administrative uses in order to carry the cost of a large computing system.

At the same time, administrators can look on micro-computers as having potential for their own applications. There are risks in this view for both teachers and administrators and we have discussed these elsewhere (Muscat & Lorton, 1981). For attendance, however, a stand alone micro-computer based system is feasible and practical. Moreover, it is an appealing application. The same question that lead to the development of the CINCH system ten years ago has lead other administrators to wonder if these small machines can't be used to provide the computing resource needed for an on site attendance system.

## DESIGN CONSIDERATIONS

The main design considerations for developing the micro-computer based attendance system were largely those arising from our earlier experience. These include: 1. Detailed Collection, 2. Automatic Processing, and 3. Flexible Reporting.

In addition, a micro-computer based system imposes some new requirements on the design. Even with a small timesharing system there is a staff of some sort. We have suggested that students might be used effectively (see Lorton & Muscat, 1976) to staff such centers. Even with a student run center, however, there is at least one responsible administrator (or teacher with administrative responsibilities) available to the school as a resource for managing the effort. With a micro-computer, this is often not the case. The system needs to be designed with both ease of operation and careful on/off-line documentation. As micro-computers are meant to exist in environments without technical support close at hand, complex applications, such as attendance, must be designed with the technical support built in.

In the case of large systems, the technical support staff exists so support and ease of operation are fairly easy to achieve. On the other hand, detailed data collection, automatic handling and flexibility in reporting are often more difficult to accomplish. With micro-computers, however, these three goals are often more easily accomplished because the size of the system means that a single application can use all the resources available (if only for well defined periods of time) with the system.

## IMPLEMENTED SYSTEM

Working from the design considerations discussed above, a micro-computer based system for attendance has been developed and implemented in such a manner that it will allow the school site administration control over the the collection and processing of attendance information with the maximum flexibility and the minimum amount of skill on the part of those responsible for the system.

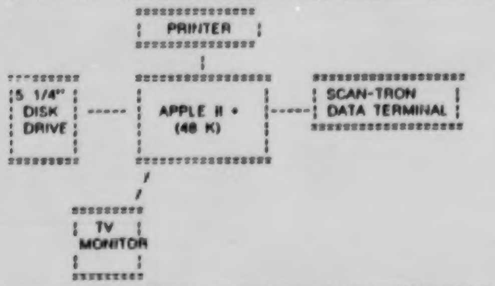
### HARDWARE

The main hardware determining the implementation is a Scan-Tron Data Terminal which optically scans a variety of forms. The reasons for designing the system around this equipment are its relatively low cost and the fact that a variety of forms exist which are designed for attendance information gathering. Further, Scan-Tron will design and produce additional forms to meet specific application requirements. The processor chosen for the initial implementation was an Apple II computer. Apple computers are widely used in educational environments and have been used with Scan-Tron equipment in school settings.

The hardware configuration is diagrammed in Figure 1. In addition to the two key pieces, the Scan-tron Data terminal and the micro-computer, a printer and one or more disk drives are used. The printer is necessary for producing the reports and the disk drives are used for holding the database and programs.

FIGURE 1.

ATTENDANCE SYSTEM: HARDWARE



Other micro-computers and other scanners could, of course, be used and probably will. At this point, however, the equipment used here is widely available in school settings and either already exists in the necessary configuration or can be easily modified to meet the requirements of the system. One of the premises on which this discussion and project is based is that school site administrators are going to want to use the micro-computers they see around them in their schools to meet administrative needs. Since a variety of micro-computers is in the schools (Tandy-Radio Shack TRS 80, Commodore PET, etc.) a system such as this must be adaptable to other equipment.

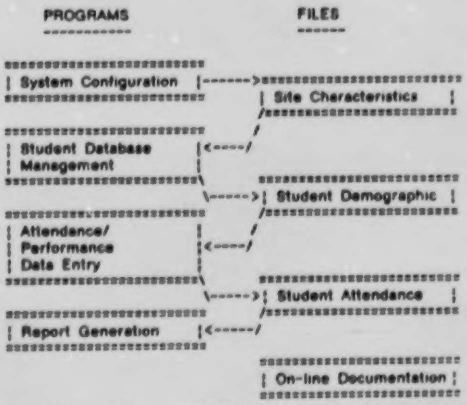
### SOFTWARE

Part of the potential for adaptation must exist in the software. Consistent with the initial hardware configuration, the current implementation is in AppleSoft Basic. Adaptation to CP/M is being undertaken in order to keep the system congruent with the belief that school site administrators will have a variety of equipment on which to press their requirements.

The software system consists of four main parts. Of these, three are used in regular operation while the fourth is used for setting up the basic configuration of the system at the school site. Because flexibility is a principle requirement and demanded by the wide range of configurations which exist at school sites, a rather elaborate configuration routine is needed. The four main parts are diagrammed in Figure 2.

FIGURE 2

ATTENDANCE SYSTEM: SOFTWARE



0. System Configuration

The variety of information schools would wish to keep in a student attendance database, the different types of attendance forms available or which could be produced, and the differences in hardware configuration are all factors which must be defined in order to install a general system in a specific location. For this reason a component of the software system was needed to set the characteristics of the site for the balance of the package.

### 1. Student Database Maintenance

Once the information to be retained in the student database is defined, it must be entered and managed. The student characteristics, such as sex, grade, ethnicity, etc., which are to be maintained, need to be entered and updated. The attendance recording is accomplished through student numbers. These are assigned by the system and then produced in a manner useable by the Attendance/Performance Data Entry component.

### 2. Attendance/Performance Data Entry

The purpose of the system is to record attendance and the main activity is the automatic handling of this information. The available Scan-Tron forms are either class roster oriented or individual student oriented. Once the system is configured to accept one of these main types of data entry, lists or forms are produced which will allow the recording of the information and its periodic entry into the attendance/performance database.

### 3. Reporting

To be useful the system must generate needed reports. In most cases reports need to list students meeting certain conditions such as all day absence, partial day absence, perfect attendance, etc. In two special cases, reports to the home and summary reports to administration, more or less detail is needed.

## ACCOMPLISHED OBJECTIVES

The five main objectives of the project were accomplished by the hardware and software described above. Specifically, the objectives were met in the following ways:

### 1. Detailed Collection

The data on attendance/performance are collected on Scan-Tron forms with the detail required by the school site. Period by period, daily attendance information can be entered into the system. The information can be collected with various levels of detail to meet the requirements of the school site administration.

### 2. Automatic Handling

Once the appropriate person at the school (usually the classroom teacher) has recorded the students' attendance/performance for the determined period, the information is handled under the control of the system. The Scan-Tron forms are read into the computer and no further manual handling is required.

### 3. Flexible Reporting

Various reports are available containing definable subsets of information. If such fields are in the student database, reports for deans, counselors, classroom teachers, parents can be produced. Other reports summarize the information by characteristics of the students. Requirements which have developed through our experience are met. While the available reporting options give the system considerable flexibility, the options are not exhaustive and some on site requirements may arise which would mean special report configurations for the individual location.

### 4. Ease of Operation

Given the micro-computer environment, the system is not difficult to operate. Because it is micro-computer based, at most sites there is a certain amount of cable manipulation and switch setting. The software is easy to use and considerable effort has been expended to make the hardware manipulation minimal.

### 5. Built in Technical Support

The technical support is contained in on line documentation information which is intended to help with the operation of the system. Extensive off line material has also been developed which will provide this support.

A main obligation of the system is that the use not usurp an instructional microcomputer. The system is maintained on floppy diskettes and can be taken on and off a microcomputer with ease. Only the time required for reading in the information and/or producing the reports will occupy the equipment. This time can be scheduled so as not to interfer with the instructional uses of the equipment.

## IMPLICATIONS AND THE FUTURE

Our purpose in this system is to develop a model which can be used to explore our basic premise - that administrators are going to evolve from viewing micro-computers as entertainment for a few teachers to viewing them (perhaps unrealistically) as substantial tools for solving a variety of administrative problems. Computing is an important tool which can be used to increase productivity and reduce costs. The application of computing in education is one of our main research interests. Our interest with this model is in finding ways of realizing the potential of micro-computers in educational administration and in determining the main problems associated with such efforts.

The low cost of maintaining computerized student attendance information at the school site will have several kinds of impact on school administration. Certain economies which come with the use of computers in the handling of information will be available to the school site administration; a feeling of "ownership" and therefore responsibility will involve the school site administration with the information being processed.

For the central administration, these school site advantages represent progress - but it is progress not without potential concerns. Two major problem areas we anticipate are those associated with distributed processing databases and with maintaining many systems rather than one system. We have built in various suggestions and procedures for maintaining compatability with central student databases but do anticipate that working through problems in this area will provide some interesting moments. Solutions to these problems will come with the use of such systems. The solutions, in fact, must come as the appeal of the systems will produce applications and use without regard to the problems generated. This is much the same case as with automated bank tellers which have increased robberies but will be used while that problem is being resolved.

The immediate advantages that school site based administrative systems have is the possible lower cost and lower personnel levels required. In addition, the on site systems can be arbitrarily responsive to school site needs. Without having to compromise features in order to meet most of the needs of several schools, all the needs of each school can be met.

Exactly what will happen remains to be seen. With the focus on meeting school site administrators' needs, the school site administration must take responsibility for both planning and operation. An advantage of even the most unresponsive centralized computing resource is that planning and operation are someone else's problem. For effective use to be made of even the most primitive school site based system, planning for its use and taking responsibility for its operation become necessary activities of the school.

We expect to be able to develop guidelines and recommendations from the operation of the micro-computer based attendance system which will help address these areas so that educational administrators will be able to make effective use of the wondrous thimble that the micro-computer is.

## REFERENCES

Harrington, R. T. Scourkas & P. Lorton, Jr. "SFUSD Attendance Data Analysis System: Design & Uses." Menlo Park, CA: CBER, 1981

Lorton, P. Jr. & E. Muscat "Formalizing student-run computer centers." Proceedings of the Digital Equipment Users Society. Vol. 2, #2. Maynard, MASS: DECUS, 1976. Pp. 519-521.

Lorton, P. Jr. and E. Muscat, "Increasing revenue to defray computer costs." Hewlett-Packard Educational Users Group Newsletter, 1974, IV, 148-149.

Muscat, E. & P. Lorton, Jr. "Personal computers in Educational Administration." Sixth West Coast Personal Computer Faire. San Francisco, 1981. (In Press).

Muscat, E.J. & P. Lorton, Jr. "CINCH: An attendance system for small time-sharing systems." Proceedings of the 14th International Convention of the Association for Educational Data Systems. Washington, D.C.: AEDS, 1976. Pp. 92-97.

Muscat, E. J. & P. Lorton, Jr. "Making CAI pay its own way." presentation for Calif. Educational Data Processing Administrators. Los Angeles, CA. Feb 17, 1978

# STUDENT INTELLIGENT TERMINAL SYSTEM

A.V. Majors
Director, Data Processing

Lloyd Westerfeld
Manager, Data Communications

Fort Worth Independent School District
Fort Worth, Texas 76107

ABSTRACT: The connecting of several more terminals to a mainframe computer used for a heavy batch work is cause for concern. Any appreciable degradation in the system's performance will have a profound affect on all users. One major goal in planning our student on-line system was to keep the additional load on our mainframe computer as low as possible. The on-line system hardware consists of a Univac 1100/10 mainframe and over 30 Tano Corporation OUTPOST II intelligent terminals. All of the software used on the intelligent terminals was developed on-site. All terminal control, data editing, and screen formating are done by the intelligent terminal. Only raw data is transmitted between the terminal and the mainframe, allowing us to use relatively inexpensive 1200 baud multi-drop lines.

The student intelligent terminal system in use at the Ft. Worth Independent School District consists of two basic components: the mainframe computer on which the student data base exists, and the micro-computer at which the user interfaces with the student data base.

The main frame is a Univac 1100/10 computer, using standard Univac software for the data base, and communications software developed at Ft. Worth Independent School District. The microcomputers are Tano Corporation OUTPOST II intelligent terminals. Software for the OUTPOST was developed by FWISD. Terminals are located in each of 30 secondary schools, the department of Pupil Records, and at the Regional Education Service Center.

The terminals in the secondary schools are connected to the mainframe via multi-dropped circuits to six ports on the General Communications System, with five drops per circuit. Each terminal has 32K RAM memory, one floppy disc, a communications interface, and a 24x80 character vidio screen. Telephone lines are 1200 baud.

## System Objectives

Intelligent terminals were selected for the online system for four reasons. First, to keep the additional load on our mainframe as small as possible. This would be a goal any time, but with 40 terminals already on the system this was a major concern. Second, to keep communications costs as low as possible. With the cost of telephone service escalating rapidly, all possible measures must be taken to reduce line charges. The use of intelligent terminals makes certain cost saving steps possible. Third, to provide immediate editing of input data at the terminal. With the intelligent terminal it is possible to edit the data as it is keyed in, perform editing functions, and return the message back to the mainframe. Fourth, to minimize conversion effort from the present mainframe computer to another computer just in case that ever becomes necessary. Software on the mainframe consists only (generally) of processing a transaction and accessing the data base to fetch and restore data.

## Meeting The Objectives

Minimize Additional Load on Mainframe - All software for generating terminal screen formats is written for the intelligent terminal, and all screens are stored on the floppy disc. All editing of input data is performed on the terminal. Only the fetching and restoring of data to and from the data base is performed on the main computer.

Keep Communications Line Costs Low - The use of intelligent terminals allows us to use fewer ports on the General Communications System by utilizing multi-drop lines. This also constitutes a major savings in that the telephone company charge per line is based on mileage. The main computer is $31.50 from the main telephone exchange, so our charge per port is $31.50 a month. However, the charge for each secondary school terminal is based on the mileage from the school to the area exchange, which in most cases is much less than $31.50. Instead of 30 x $31.50 per month, the cost is 6 x $31.50 plus the cost from a school to the exchange.

Additional savings are realized by using 1200 baud lines. Faster baud rates, at higher costs require more sophisticated modems at a higher cost. If an entire screen, in this case 1920 characters, is to be sent down line you, the user, would want higher baud rates to fill the screen faster. (At 1200 baud, it would take about 16 seconds to fill a screen. Since we format the screen from the micro-processor, and receive only "live" data from the mainframe, it takes only 1.7 seconds to transmit data down line).

Provide Immediate Data Editing - As stated earlier, all data editing is performed on the terminal, with instructions written for the micro processor. There is no waiting for data to be shipped down line, edited and an error message returned. Numeric fields must be numeric, ranges must be met, required fields must be filled and all fields must be complete; otherwise, the user is not allowed to move to the next field. User errors are recognized immediately and must be corrected before the next step begins.

Minimize Task of Conversion to Another Mainframe - We all know that sooner or later our present smooth running system will have to be converted to other hardware, which usually means another mainframe. Software for the Student Intelligent Terminal System on the mainframe has been held to a minimum. Only the transaction processing component, that part that recognizes a transaction, and fetches and stores data, will have to be replaced.

## The Terminal Software Components

One component is the monitor, made up of the initialization code, the communication package (which emulates Univac's Uniscope 100 terminal protocol), and a set of macros which perform basic micro-computer functions. These functions include screen-to-memory data moves, memory-to-screen data moves, memory-to-memory data moves, get a keyboard character, display a character, position the cursor, get the cursor position, write to disc, read from disc, perform data compares, and others.

A second component is the pre-written screen formats which are used to create the screen displays into which data is placed. These are developed with an editor written specifically for this purpose. This allows one to easily describe the screen format, then verify and display the format. The formats are stored on the mini-floppy disc.

The third component is the Driver, or the program which directs the functions of the intelligent terminal. Once the program is initiated, the terminal takes over and the user responds to the terminal requests. The user makes decisions about which screen to use next, what data to enter or update, and when to close the terminal.

The programs within each of the three components were developed using a compiler level language written for the micro-processor and macros developed by Ft. Worth ISD in assembly code. The compiler allows us to develop programs on the 1100, then dumps them down line to the micro-processor. Programs can be developed much faster on the mainframe than on the terminal.

## Screen Format Build Instructions

A screen format build program allows the user to build, test and alter screen formats for the OUTPOST II. The program allows for easy and quick changes to be made to the screen to determine the best layout possible. By becoming familiar with the program capabilities, one can quickly create many different screens.

There are seven types of functions that can be performed. These are Create, Display, Edit, List, Pack, Read and Scratch. Each type function is explained below.

### Create

The create function is used when one wishes to build a new format. One requests the create function, and is placed in the create mode.

The data is entered one line at a time. One line may describe from one to 80 characters of a screen, with those 80 characters all being on one row of the screen. As many lines as needed may be used to describe one screen row. Each line of data is terminated with the "RETURN" key. One may alter the data line any time before the "RETURN" key is struck by using the "RUBOUT" key. After the "RETURN" key is struck, that line of data is moved to the output buffer and also placed on the screen above the input area. Then the next line of data is entered. After the complete screen format has been entered, the "ESC" key is struck, and the screen format is terminated. An output name is requested, and the screen format is written to disc. Then it is read from disc and the screen described is displayed on the terminal screen. The screen display is done in the Read Mode which is described below.

### Display

The display function is used to display the disc format directory on the terminal screen. It displays the name of each format and the relative disc location of that format.

### Edit

The edit function is used to make changes to an existing format. The format may be requested and displayed on the screen. It is displayed in line form. The cursor positioning keys, the character insert and rubout keys, and the scrol key may be used to alter the data on the screen. Upon completion of the editing, the "ESC" key is struck to terminate the editing. At that time the format name is requested and the same name or a new name may be given to the format. If the same name is given to the format, then the old format name is erased from the directory. If a new name is given to the format, the old format remains accessible to the user.

### List

The list function is used to print the format requested on the printer. After requesting a list, the user will be reminded to turn on the printer and ask to enter the format name. Upon entering the format name, the format will be found and printed.

### Pack

The pack function is used to make it possible to reclaim space on the disc that had previously been used for formats that have been changed or deleted. When a format is changed, the new version is written to the next available location on disc. Since the old version is no longer available, that space becomes available. Also, if a format is no longer needed, it can be deleted. While this space is no longer being used, it is not available until a pack is performed.

### Read

The read function is used to test a format. The format data is used to display the screen that is desired. If an error is detected in the format, the read routine will discontinue the display of the format at that time and indicate to the user the location of the error in the format. The user may then use the edit function to correct the error. The read routine will then be used by the edit function to display the screen again. This procedure can be used until all format errors have been removed and until the desired display has been achieved.

### Scratch

The scratch function is used to remove a format from the disc that is no longer desired. It simply removes the name of the format from the format directory. After a pack has been performed, the space is available to the user.

By using the functions described above, one can enter the data necessary to create a screen format which will be used by the online system to build screens. The following commands may be used to describe the screen.

### S(x,y)

This command indicates where on the screen this screen display is to begin. The values assigned to x and y will determine the starting coordinates on the screen. The value of x must be in the range of 0 to 79, and the value of y must be 0 to 23.

### Px(display)Dy

The "P" command creates a protected field. This protected field has a length of x, where x is less than or equal to 80. The brackets and the data within the brackets are optional, and the data will be displayed in the protected field left justified. If no brackets are included, the protected field will be spaces. The Dy portion of the field is optional. If used, Dy points to data to be displayed in the protected field.

### AxDy

The "A" command creates an alpha field. This is a field of length x, where x is less than or equal to 80, and where only alpha data can be entered. The Dy portion of the field will be used to indicate the location to which data entered is to be stored or from which data is to be obtained to be displayed at this screen location. This command will be more fully defined at a later time.

### NxDy

The "N" command creates a numeric field. Otherwise, this field functions exactly like the "A" command.

### BxDy

The "B" command creates an alpha-numeric field. Otherwise, this field functions exactly like the "A" command.

### DxDy

The "D" command creates a special-character field. Whatever characters are specified as acceptable special characters by the application may be entered in this field. Otherwise, this field functions like the "A" command.

### ExDy

The "E" command creates an alpha-special-character field. Otherwise, this field functions like the "A" command.

### FxDy

The "F" command creates a numeric-special-character field. Otherwise, this field functions like the "A" command.

### GxDy

The "G" command creates an alpha-numeric-special-character field. Otherwise, this field functions like the "A" command.

### HxDy

The "H" command creates a special field which is protected but marked as an updated field. This data is therefore automatically sent to the 1100 as control information. Otherwise, this field functions like the "A" command.

### Dy

The "Dy" subfield is a relative pointer into the data record received from the 1100. This relative address is also returned to the 1100 with any updated data field.

C(x,y)

The "C" command specifies the initial location of the cursor after the screen has been displayed. This command should be the last command of each format. As with the "S" command, x and y indicate the coordinates.

Ex

The "E" subfield specifies the edit function to be performed on this data field. Following are the types of edits available.

E1 - 0-12 Month Only
E2 - 0-31 Day Only
E3 - BG Sex
E4 - ACDW Course change code
E5 - ABCDEFSNUI Grade of citizenship
E6 - 0-16 Withdrawal codes
E7 - B1B2C1C3 Enter codes
E8 - 0-6 In-eligible reasons
E9 - BHIORW Race
E10- E I SP Bus data
E11- Special Ed codes
E12- 1 SP Parent Military
E13- CE1234567890 Shots dos
E14- DVTMRH
E15- 1-2
E16- 1234567F
E17- 01234567
E18- 5 0 0
E19- HAB
E20- 000102030405062233404144 4555
E21- 01-32
E22- W SP
E23- GRADES

Sx

The "S" command specifies that a special edit function is to be performed on this data field. Following is a list of special edits available.

S1 - Edit the course # based on school master section
S2 - Edit the section # based on the school master section
S3 - Edit the semester code based on the master section
S4 - Edit Birth Date
S5 - Edit School Date
S6 - Edit Month
S7 - Edit Day
S8 - Edit Attendance Cycle 1
S9 - Edit Attendance Cycle 2
S10- Edit Attendance Cycle 3
S11- Edit Attendance Cycle 4
S12- Edit Attendance Cycle 5
S13- Edit Attendance Cycle 6

J

May be used only with numeric fields and causes the data entered to be right justified.

Mx

Sets the display mode of the following protected data. The following values result in the following display modes.

0 - Normal video
1 - Blink
2 - Invert
3 - Invert,Blink
4 - Grey
5 - Grey,Blink
6 - Grey,Invert
7 - Grey,Invert,Blink

R(AAA)(XXX...X)(Xx(...KZ)

The "R" command is used to solicit the information necessary to obtain Data Base data required for the screen. "AAA" is the 1100 transaction code. XXX...X is the request message. Xx specifies the type of information needed. The "KZ" is used if special editing is required for the screen, where Z is an "A", "B", or "R". Multiple items of information may be requested by repeating the request portion of the field.

U(AAA)

AAA is the name of the screen that will be displayed if the "First Screen" key is depressed while this screen is being used.

V(AAA)

AAA is the name of the screen that will be displayed if the "Next Screen" key is depressed while this screen is being used.

W(AAA)

AAA is the transaction code transmitted to RDC with the data entered while this screen is being used.

The commands must be separated by commas. Each field of the format must completely reside on one row of the screen. However, as many format lines as desired may be used to describe one screen row. A "." may be used to terminate a screen row. The use of the period will result in a protected field of spaces. The "RETURN" key and its display must be the last character on each format line. The "ESC" key is used to terminate a format.

Using The Terminals

The intelligent terminal system may be initiated by turning the terminal on or by pressing the load switch. The program and selected editing data will be loaded from disc, and a password will be requested. Upon receiving the password, the terminal will send an open request with the password to the mainframe. If the password was valid, the mainframe will send a school # and special editing data request to the terminal. If not valid, the password will be requested again. After the requested data has been received from the mainframe, the terminal will display a menu on the video. The user may now select the type of student data desired by requesting the appropriate screen. The student ID will be requested, and then the request will be sent by the terminal to the mainframe and the screen will be displayed upon the video. Upon receipt of the student data from the mainframe, the terminal will place the selected data in the screen on the video. The user may read or alter that data in any way. Upon completion of the use of that set of data, the user may make another request.

## Conclusion

Micro computers are being used in almost every facet of education and industry. Within the Ft. Worth Independent School District, OUTPOST terminals have been used by students in computer math classes as interactive terminals. Our maintenance department has used the terminal to develop energy conservation systems. Other uses, in an on-line and stand alone capacity, are under consideration.

213

# INFORMATION SYSTEMS MANAGEMENT

Herbert C. Mayer
Department of Management
and Administrative Systems

Washington State University

Management of the data processing function within an organization is traced from the early days of data processing through several generation changes to the present. A number of problems for management concerning the changing nature of this activity over the past 25 years are discussed. The impact of new technological developments, the level of recognition within the company of the computer activity, the implications for top management of a Management Information System, the adequacy of service to users, and the effects of the minicomputer are analyzed. Finally, the impact on education of the minicomputer is discussed as it applies to both large research institutions and small colleges and school systems.

As computers have grown in importance and in their contributions to an organization, the problems of managing the computer activity has increased in complexity much faster. To understand how the management process has developed over the past twenty-five years, let us go back historically to the way in which most companies started with data processing.

## Data Processing Management: The Early Years

The punched card was used as a time card. As workers came to the plant they passed the time clock which contained a pocket large enough to place the time card in and by pushing the card down the starting time would print on the card. The process was repeated on the way home, so that the total hours worked could be determined from the start and stop times. Cards were collected at the end of the week or pay period, and sent to keypunching, where they were read and punched in preparation for payroll calculations. This activity of time card processing was under the management of the financial function usually delegated to the Payroll Department. Thus, we see that the financial part of the organization already had a very practical and necessary application that could be converted to a computer application whenever the firm took the step to acquire a computer. The direct management of the acquired computer generally was placed in the hands of a financial manager, who had limited knowledge and exposure to the computer. He delegated almost total authority to an individual who had familiarity with the application to be computerized, i.e., Payroll. We see even in the early days that management had some concerns about this new equipment and tended to stay aloof from the actual operation and activities surrounding its implementation. Data Processing, as it was called for many years, was usually buried in the financial part of the organization. It was then evident that most of the early applications would be related to the accounting function with little interaction with other groups within the firm. It is true that some attempts were made to apply some numerical analysis techniques to the solution of differential equations by a few of the scientists in research and design activities. By and large, control of what type of applications would be run on the computer remained in the hands of the financial people for a number of years.

Management concerns were concentrated on improving the process and time required to complete some of the basic accounting applications. In the case of the Payroll application, with mostly manual steps involved, the process was taking almost the total time available from Monday afternoon to nearly Thursday midnight to prepare the weekly checks and associated records and reports that had to be maintained. Other applications could be run if they could be squeezed in between the various steps required to complete the payroll processing. Data processing duties rested in the hands of a relatively few people who had training in punched card techniques and knew how to wire boards to accomplish the variety of steps that each application required. The major problems encountered in those days included the protection of the completely wired application, having enough boards to take care of new applications, the training of people to conduct this early form of application development, and communication between this technical computer group and both management and users.

The data processing installation consisted of a number of assorted pieces of "tab" equipment, each of which performed a separate function or group of functions. The board that was wired to program the different functions required for an application was placed in its holder on the machine in order to make the proper electrical contacts. Because it took considerable ingenuity to wire each board to accomplish the desired set of functions, there was an easy temptation to go back to earlier-developed boards to repeat or borrow a group of steps that were needed for the new application. This borrowing process occurred frequently because much time was invested in the development of prior boards and a large portion of that time could be saved while working on the new application. Because of this activity to go back and re-use previously wired boards, there were frequent shortages of boards available for new use. It also appeared that the delivery time was so long that people thought twice about going through the purchasing process unless absolutely necessary.

The training activity took one of two forms. If there was sufficient planning time, individuals could be sent to classes conducted by manufacturers in these board wiring techniques. However, as was frequently the case, the time was not available so a person had to pick up the knowledge and skill through on-the-job training, where a senior staff member served as an advisor and teacher. Obviously, this second route was less efficient, but provided the required resource when needed. The communications problem developed through the difference in technical backgrounds between the staff and management. Somehow communications concentrated on the operational problems encountered which were not well understood by management. The ability to get the work done was impaired by the failure to understand the true nature of the problems. Management tended to stay apart from the D. P. activities and let the operations staff run the shop to the best of their ability. Another drawback for the D. P. staff was their lack of knowledge about management concepts and how data processing fit in with the rest of the organization. Both of these limitations tended to reduce the effectiveness of the data processing activity, and thus it took a long time for the D. P. organization to be recognized for what it really could contribute to the firm as a whole. This lack of recognition also hindered the contacts with users who desired an application developed. Some groups felt that even though the computer might provide results faster, they could continue to rely on manual methods and use of a desk calculator, which they understood and could be certain of the accuracy of the results. Even if the manual approach took longer, the user was certain of the correctness of the figures obtained, whereas results from data processing were of unknown quality. Again, D. P. had a significant problem to develop its image with all users before gaining wide acceptance in the company.

Almost all of the work of the computer consisted of applications that had been in operation manually, time could be saved, and the number of people involved in keeping the manual records could be reduced. As we think of computer applications and activities today, there was certainly no thought given to a Management Information System back in those early days. It took many years, a wide penetration of computer usage throughout the organization, and a more greatly informed management to realize that the possibility of an MIS for the whole company was feasible, or had potential use in assisting management in making decisions.

### Data Processing Management: Growing Years

As the role of the computer assumed a greater importance to the organization, management realized that they must take a firmer hand with the data processing activities. After the implementation of the first few accounting applications on the computer, financial management had already seen the impact of the computer in the financial area. Much of the computational load had been transferred to the computer and the need for the large numbers of accounting clerks had been reduced. Now the computer had a management advocate other than the technical people who operated the computer and developed additional applications. The push was on to develop many of the other routine accounting applications that also required large numbers of clerical personnel and had common elements with the applications already implemented. Nolan (1978) called this the stage or rapid program proliferation and development. During this period management encouraged wide-spread use throughout the company and hurriedly hired and trained a computer programming staff to assist others in developing new applications. There was also a greater awareness on the part of financial management that the computer had a definite place in the company and that they should help to inform other management about the capabilities of the computer.

One of the groups that needed no indoctrination was that in the technical areas where sizeable computations were quite routine for design, engineering and other research activities. As a matter of fact, some computer advocates in research had requested time on the computer. Those companies that felt the computer activity belonged to the financial function sometimes found themselves operating two computers, where one was devoted exclusively to the accounting routines of the financial function and the other was operated for the benefit of the scientific users who performed research, development, and engineering activities. Each group had an entirely different computer because computers twenty years ago were made specifically for either the scientific type of calculations or the data processing "large volumes of output" applications. The internal capabilities of each type of computer were considerably different, with the business computer emphasizing words and fixed point decimal arithmetic, while the scientific computer provided significant computation ability for both fixed point and floating point calculations. The business machine had a wide variety of input-output handling capabilities including those for large files and the printing of a variety of reports.

The management of some firms assumed the computer facilities would serve all functions and groups with the company. This approach resulted in having one computer installation which might contain more than one type of computer that could serve both scientific and business users. The structure of the management organization of the computer department began to include development, service, and technical support functions in addition to the operations arm which kept the computer going through one to three shifts each day. A little time was spent in educating others about the computer's capabilities so that new areas could be explored for possible applications. Management concentrated on keeping and attracting new qualified personnel to work with the computer. Most efforts were concentrated on keeping the computer operating with its increasing load and handling the conversion to a newer, faster computer with vastly improved capabilities. So much of the total department was involved in the operations side that development work frequently took a back seat. It should also be pointed out the conversions took place every few years and sometimes the total data processing department resources were utilized to accomplish the changeover. In many of these conversions, changes involved a totally new operating system, new programming languages, new type of computer and a new education process for computer personnel as well as users. Where there were major changes of equipment and programming languages, all of the current computer programs had to be reprogrammed and this was a major effort involving everyone.

It became clear to management that frequent conversions were very costly and could set back, temporarily, the current activities that had to be handled. It was also recognized that the cost of the computer department was increasing much faster than other costs in the financial function. Some controls must be instituted in order to keep these costs from getting out of control. While operations must be continued, new program development could be slowed at the risk of some dissatisfied users. However, with controls in place, users would be paying more of the data processing costs or at least sharing in these payments depending on the amount of equipment and service used. Included with these controls was the beginning of a priority system to establish which applications would be programmed next. But the setting of priorities was impossible for data processing management to determine, so two techniques were established to handle this problem.

One method was to put this decision activity in the hands of the computer advisory committee which had many representatives from the user departments. Such a group was better able to handle the setting of priorities because each department was generally represented. Where organizations had no comparable committee, the problem was usually pushed to a higher management level for appropriate analysis and decision.

Because of the spread of computer activity within the firm, management recognized the growing importance of the data processing department throughout the entire organization. It was realized that in order to achieve company-wide acceptance, the data processing department must have higher status within the organization. Thus, many companies took the department out of the financial function and gave it status at the vice president level and created the position of Vice President of Information Processing. This change gave recognition to the concept that "Information is a company resource." At the same time that the level of D. P. within the organization was raised, greater overall responsibilities were given to this important function. Total control over the acquisition of computers was a part of the D. P. charter. Any investment by the firm in possible new computers had to be reviewed by the Information Systems Department. A further organization change was made to include an Information Systems Office within the corporate organization, an additional recognition of the importance of information to the company. More particularly, these changes meant the development of better computer programs for the collection and analyzing of a variety of types of data for later use by management. Such was the real idea of a comprehensive Management Information System, which could provide the appropriate information needed to help in making decisions at all levels of management.

## Data Processing: Organization Structure

Along with these greater responsibilities, the MIS department had also made many organizational changes. The department now included a sizeable operations staff that ran the computer and peripherals around the clock. Provisions were made for a systems staff to keep the software operating system or executive program up-dated with the latest changes and capabilities that the company required. This group also kept abreast of the latest computer technical developments both in hardware and software so that the firm could keep their facilities technologically current. The demands for computer usage had increased many-fold in the previous decade and a special staff was set up to handle the variety of ways of preparing input for the computer and to distribute reports to the many users as they came off the printers. The need for a development group had existed for a long time, but it was in this area that reductions were made whenever the department had to cut back. A considerable backlog of new programs to be developed had built up over time, so that the users, who were waiting for new applications to be developed, finally decided that they would have to undertake their own development work if new applications were to be realized. Thus we see the trend toward development of application by users outside the computer department. This change in functions performed by D. P. is probably a very healthy change for the organization because it gets the user heavily involved in a desired application and places the major responsibility for that program in the user's hands. Thus, D. P. provides a true service of making the computer available to anyone (who is willing to write a computer program). Some limited development work continues to be a part of D. P. activities, although it tends to be concentrated on system changes or the testing of new programs that may be added to the total operating environment.

Another sizeable activity that has grown tremendously is that of maintenance of existing programs. Perhaps the largest expenditure that D. P. departments are experiencing today is that related to correcting problems in programs that are supposed to run correctly but don't. Significant programming activity is maintained by most computer departments to handle just this program maintenance capability. The reason for this situation is that the D. P. departments in the past had the staff to develop most of the programs that are currently in use and therefore have an obligation to correct any deficiencies in these programs whenever they are discovered at a later time. The maintenance activity also holds true for special application programs that have been purchased. Frequently such programs are acquired "as is", which means that the user has to make them work and the seller has no further obligation to maintain or make modifications. Purchased programs have become a real headache for many D. P. departments because of maintenance. The only word of caution that can be given is that each potential procurement of an application be fully investigated to make sure the terms are satisfactory to the buyer.

For many large D. P. departments there are two additional required functions in the organization. One is an administrative staff to help handle the load of activities required to keed records of all personnel, operations, and costs. Perhaps the most important function here is to determine the cost of operation and an appropriate charging algorithm that is used to assess charges to each user of the computer facility. Obviously, the method developed should reflect corporate or financial policy on the level of user support. A second D. P. function that has increased in importance is the need for constant education and training. The training group provides the education of all the current D. P. staff about the operating system and functions and issues documentation about new or proposed changes in the system. In addition, they help to keep the technical staff aware of educational opportunities for short courses, seminars, technical meetings and other related activities. The training staff also has a continual activity in running classes for the operations staff and the training of new people. Turnover is a constant problem in operations because it is probably the largest in the D. P. department. Where the training staff is large enough, they have a responsibility to educate new users and update current users. With the trend now to involve users in developing their own applications, the training staff can hold classes on an "as needed" basis. Constant involvement with potential users not only provides education about the D. P. facility but also helps improve communications with the users. Newsletters offer an additional means of informing users about computer activities.

Adequate user involvement with data processing has been difficult to encourage. A frequent method has been to establish a computer advisory committee containing representatives from the various departments or functional areas within the organization. Each representative can serve as a communicator to the home department about D. P. plans and changes. At the same time these people can give the data processing department considerable input and feedback from the users about the adequacy of their service. A committee of this type, when used for the benefit of everyone, can serve as a very satisfactory sounding board and an excellent means of communication with users. For those D. P. departments that are interested in feedback from the users, some groups have developed a survey report card to indicate how well D. P. is serving its users. Such techniques help to develop better relationships between the computer staff and the users. For those organizations that are either smaller in size or do not have an advisory committee, some D. P. departments have set up the position of user-coordinator. The main duty of this person, who may serve either full-time or part-time, is to keep regular contacts with the users and know what their needs are. If anyone is encountering problems in getting good service from the computer, the user-coordinator can help to clear up the problems. This individual gives the users someone to talk with directly to get problems solved.

## Recent Trends in Data Processing

Some of the medium-sized and large computer installations have moved to distributed processing. A second trend shows a movement toward time-sharing which provides computer capabilities to many users through display terminals. A third trend has been the impact of the minicomputers on users and the main computer facility.

Distributed processing permits the computer installation to schedule job runs at either the central facility or at an outlying smaller computer having sufficient capacity to execute the specified jobs. This distributed activity offers the user computer capability near the work location of the user, who can run many of the smaller jobs at the remote site and when necessary send data and programs to the central facility for execution. Scheduling of computer facilities becomes a more essential part of the operations activity. Another user advantage is that the testing of new programs and changes in old ones can be handled relatively close to the work location with a resultant saving of time. Once again the systems staff of the central facility have to make the necessary changes to the operating system so that the remote computers can send and receive data and programs from the central site. Many remote facilities have considerable computing power capable of running a number of programs without involving the main computer. Only when the larger memory and computer power are needed is it necessary to transmit requests to the central facility.

Timesharing and inexpensive terminals have made possible direct use of the computer by all in the organization who want to use the computer. Through a program that regularly checks the status of each terminal connected to the computer, it is possible to perform your own calculations in the privacy of your office. At the same time, those who have the appropriate key-word identification can make inquiries of centrally kept data bases. This capability has made considerably more information available to more parts of the organization instead of relying on printed reports. The computer also shortens the time span between source input and availability of information. There are organizations who have reports on the total sales activity for the previous day, making it possible for management to concentrate on those areas that require attention. Such prompt action was not possible before the days of computers because the data collection process was manual and it took a long time to prepare reports. When the reports were available, the problems had become more pronounced because of the considerable lag in time. Action came so far after the fact that it took much longer to correct the problem.

The availability of a terminal to do computer work meant that much time could be saved by the user. The display terminal became a part of the daily activity in production, since it was available any time of the day to check the current status of any step of the manufacturing process. It was possible to determine how well the process was going and where there might be difficulties. Immediate attention could be focused on those areas to attempt to rectify the problem and keep the process moving along. Individuals used the terminal to develop new programs and had instant results for the executed programs. This capability helped to bring down the time taken to develop programs and produce results. The only drawback here is an informed management, who would recognize that providing such facilities really increased the productivity of individuals.

The trend toward the wider use of minicomputers and micro-processers comes from two developments. Advances in technology concerned with the miniaturization of electronic components have accelerated the development of smaller and more compact computers. Along with this change, the price of the components has dropped drastically, so that it is now possible to acquire what used to be a medium-sized computer for just a few thousand dollars. Applications for these computers were available by modifying them to fit within the memory capacity of these small computers. With all of the computer's power provided through hardware and software in a small package for a relatively inexpensive price, it was now possible for researchers and other users to acquire their own computer provided they had the necessary funds. These developments also allowed the small business entrepreneur to purchase and run a computer that fit the needs of the small business. This spread of usage provided another problem for each of these computer installations -- the increased demand for computer-educated people to run the computers and develop additional applications. The increased number of computers raised the total needs for both computer science students or business students taking a computer option so that the two areas could be blended into a very appropriate background for work in the business world. Such demand is expected to continue for a number of years as the smaller computer makes extensive inroads in all areas of business, government and education.

The rapid spread of small computers raised questions in many of the larger computer installations concerning their policy of control over all computer purchases. For a number of years a major means of control over the computer activity had been to require the approval of a central group for any additions or new computer equipment. Along with this centralized control of both computer power and the purse strings has been service to all users. An acceptable level of service in the eyes of the users, provided satisfied users. Under these circumstances, control of the resources usually kept services at an acceptable level. However, this situation did not occur in all companies, and dissatisfaction grew as it became more apparent that the computer department was less and less responsive to the needs and requirements of the users. Particularly in the research sections of organizations where there was control over their own funds, it was possible for researchers to buy inexpensive computers to be used in their research efforts. The fact that they could also perform many of their routine calculations on the smaller computers and do it at less cost than the charges from the central facility, made them happy, but frustrated the computer management that had fought so hard to keep this from happening. This developing trend of acquiring many small computers throughout the firm is a situation that management must accept or improve their service. The problem is very deep-seated because it also concerns the level of satisfaction of the service received from the centralized computing facility. It will not be solved without some real modification to current corporate computer strategy. This problem poses one of the real challenges of the '80s for information systems management.

## MIS Implications

During the years of growing computer capability within an organization, it was gradually realized by management that the computer was a valuable tool not only for the operating level of management and researchers, but also for higher levels of management as well. The change in attitude by management toward the usefulness of the computer came very gradually. The process of educating top management came from a push by data processing management and vendors who provided computer education for executives that included numerous ways in which the computer could provide top management with information useful in decision making. Also, a few of the early managers of computer installations had progressed up the corporate ladder to positions of fairly high importance within the organization. The influence of each of these individuals could be felt in the types of information that was requested because they had familiarity with a number of the applications that had been developed some years before. This intricate understanding of certain applications was a key element in the recognition that data collected concerning the daily operations of the firm could be processed in such a way so as to provide useful and significant information that helped in short and long-term planning. So, many managers with limited experience with the computer had to accept the word of colleagues about what information could be furnished. The knowledgeable former D. P. executives became both the source of ideas and the experts on making the computer work for management in very meaningful ways. The concept of the corporate Management Information System, that could provide top management (as well as other levels of management) with more accurate, more timely and better organized information for consideration in strategic planning, became a reality.

The great variety of individual applications in the financial, manufacturing and marketing functions of the company served as an excellent starting base for this MIS activity. As additional programs were developed, they could fit into the structure of an enlarged information system, because most applications contained a large share of common elements with existing applications. It was also the desire on the part of the computer department to try to get some consolidation of source data, that was repeated for one application after another. The resulting reduction in duplicate use of memory could save considerable space for additional applications.

In many respects, the feeling exists among computer users that MIS is a computer program or collection of applications that serves the entire needs of all levels of company management. This myth should be exploded right now. MIS is a conceptual system which contains a number of components, one of which would be the collection of computer applications related to operational activity. Another component is made up of the statistical, mathematical and special computer program tools that assist in analyzing data, provide simulation models, and develop forecasting techniques used by many company analysts and researchers. A third part is made up of the large body of data collected from the environment about the industry, competitors, suppliers, the economy, government regulations, etc., most of which is not quantifiable and therefore is kept outside the computer. This total information system should therefore provide just about any combination of information that is needed by operational, tactical or strategic levels of management. In the computer portion data can be combined from various applications and data bases and processed into a summarized form for the higher levels of management. Probably the most important feature for top level management is the comparison of information against both past performance and projected forecasts. These comparisons make clear immediately how well a company is meeting its planned goals. As rapidly as information is provided for the executives, they will place even greater demands on alternate or additional types of information required. Thus, we see that once the ball has started rolling it is almost impossible to stop it - i.e., further requests for more and more useful strategic level information will increase.

## Information Systems Management in Education

For the education environment, it is best to divide the area into two groups, one for higher education and a second for school systems encompassing elementary and/or secondary schools. Many of the problems in the computer departments in higher education, both colleges and universities, have similarities to companies with medium-sized computer facilities. However, in education the emphasis on computer use is focused on three areas: instruction, business or administrative applications, and research.

A research-oriented university typifies the educational institution that has strong resemblance to many companies. For many years there has been a central computer facility which attempted to serve the needs of faculty, administration, and students. Significant numbers of faculty members have utilized the computer for a variety of research activities. For this group the central computer facility maintained package programs in statistics and mathematics which could be used easily. The computer system also provided sufficient storage for data files so that it was relatively inexpensive to leave the data on the system ready for use. The computer facility also allowed the development of new programs for specialized research analysis, so, together these capabilities served the needs of researchers. The second area of business applications included the typical financial and accounting routines similar to those in business, but in addition there were programs needed to handle some of the special procedures in a university. These included registration for classes, student sectioning algorithms, the handling of admission records, grade-keeping, library acquisitions and circulation control, student accounts, student housing, book store activities, and many others. Generally, there was adequate support for the development of these programs whether the staff was located in the computer department or scattered throughout the campus in various administrative offices. Support was provided for the daily and routine programs, while others waited on a priority basis to be implemented. This approach is very similar to the one taken in industry.

Instruction seemed to fall in a special class by itself. Support for this area varied from school to school depending on where each school set its priorities for these three computer application areas. You will find a few universities that have placed a high emphasis on student instruction and at these schools there are practically no impediments to the opportunity for each student to get as much computer instruction as desired. Obviously, as the instruction priority falls behind the other two service areas, the quality of computer instruction the students received was considerably reduced. There are many considerations that go into determining the priorities to be established, with cost being one of the current funding areas that continues to affect the decision. By and large, colleges and universities have used the existing computer facilities to provide some computer instruction.

## Impact of the Minicomputer

The small computers available at very low cost provide an answer to the ability of a school to make available instruction on the computer. This is very much the case for both school systems and smaller colleges who have not had computer facilities. In larger universities, the inexpensive computer did not come to fill the instruction need. Researchers gradually became disenchanted with the central computer facility for two main reasons. The cost of services received tended to rise over the years, and in numerous cases users had the feeling that they were paying for facilities that were seldom used by them in their applications. Thus, they were supporting facilities they did not use and they felt they should not have to pay for. The second area concerned the insensitivity of the main computer staff to the needs and concerns of academic users. For whatever reasons, users found it very cumbersome to utilize the central facility and help was not easy to locate. Added to these concerns was the technological development of the inexpensive small computer that any researcher with sufficient funds could buy, without having to clear the purchase with a central computer committee. Through this avenue, researchers have been acquiring computers at a tremendous rate and thus we see the great proliferation of the minicomputer and microprocessor. These are used almost exclusively for research and computational purposes, with limited instructional use.

As the small computer began to affect the central computer facility, computer management realized that they now had some competition out among the users and they needed to improve their communication with the users and the service they provided to avoid further loss of computer usage. It should be pointed out that the recognition of changes among the users did not always awaken the data processing management to the fact that changes were in order. Some facilities were willing and able to make changes to accommodate the great majority of users. Others continued current policies because the bulk of the usage was in the administrative area with a captive audience, so there was little danger of losing them. Unfortunately, this attitude took its toll on the instructional usage and the losers were students who had limited accessibility to the computer. Most of the above remarks apply to that segment of universities who have a heavy investment in research facilities and outside sources of funding to underwrite these activities. An additional trend by these institutions has been to acquire a small to medium-sized computer for an entire college or even a department. Such acquisitions usually required a combination of grant money and university funds in order to purchase and maintain the equipment. By and large the computer served the interests of the faculty and students in that one area, either a college, such as Business, or a department, such as Chemistry. With the initial investment largely funded from outside sources, the college or department can operate the computer to suit themselves and have just the smaller costs of current operation to pay. Over a period of time, the total cost tended to become less than buying time on the central computer facility.

## Small Colleges and School Systems

Considering the greater number of colleges and universities that do not have the heavy emphasis on research and investment in facilities, there continue to be efforts to obtain minicomputers, but because of the funding problem, these attempts are coordinated with the central computer facility. Support for this type of acquisition comes from a cooperative approach to the administration to request funds. Many schools desire to have the students learn using the latest equipment and the types of equipment that will be encountered by students when they go out into the working world. Funding is the major hurdle that must be overcome. Some schools have placed computer instruction first among their computer activities and where this situation exists, the students are receiving very good training in the use of computers. The important concern here is the attitude and actions of information systems management in providing a good working atmosphere for all users of the computer. It can mean that computer management should investigate the latest technology and acquire some minicomputers so that computers of different sizes are available to students and faculty. A greater awareness of current trends and developments considered within the current budgetary constraints will demonstrate to users that the desire is there to help provide as close to the future working environment as possible. Some of these considerations apply equally well to the administrative area where word processing equipment is being used increasingly and can be a life-saver in the preparation of reports, articles, memos, books and papers.

When school systems are considered, we can see that the nature of the computer management problems is but a subset of those encountered in large installations. Usually there is a relatively small or medium-sized computer facility that serves the business office of the schools and attempts to handle instructional use. The teaching of one or two programming languages in relation to either a math or a business class is a frequent occurrence. The people to be satisfied are very few -- several teachers and the business staff. The management structure is easy to determine - one person in charge of the facility with just a few users to keep happy. Obviously this is an oversimplification because very few computer facilities remain in this type of framework. Greater demands are made by the business office and instruction gets shortchanged on available time. What we see developing is the constant problem of growth. Every time a new person gets involved, there are more and more requirements. Even management can get into this tug-of-war act as information is supplied that is helpful, additional requests can be made for more information. The capabilities of the existing computer are outgrown and now management must make a decision to replace, upgrade, or add to the existing facility. When the decision is made to make a change, there is the trauma of going through the conversion process, which is never easy. After the dust has settled and the change completed the battle between users, priorities, and management starts in again (at an insignificant level for a period of time). And the cycle starts over. There seems to be no end to the process!

In summary there are constant problems of change to be dealt with in computer management. The name has been changed from data processing management to information systems management. The changes in technology come so rapidly that the operation of obsolete equipment is a continual problem. Hardware costs have dropped so much that the costs now center on the people in the organization. Miniaturization of components has made possible the minicomputer and the microprocessor, which now allow individuals to develop and run their own programs. Management's problems continue to focus on getting the work out in a timely fashion with up-to-date equipment that satisfies all the users and provides an appropriate environment for computer professionals.

## Reference

Nolan, R. L. Organizational Response and Information Technology. AFIPS Conference Proceedings, Vol. 47, National Computer Conference, June 5-8, 1978, AFIPS Press, Montvale, N.J., p. 517-524.

# A COMPUTER DEMONSTRATION ROOM FOR EDUCATORS

Lawrence Molloy and Ed Herstein

Wayne County Intermediate School District

Wayne County Intermediate School District has traditionally offered area schools both administrative and instructional computing services through a network of minicomputers and a mainframe. During the last couple of years, a growing number of local educators have become actively interested in doing instructional computing on microcomputers. The Intermediate has enthusiastically promoted this interest by creating a four-member Instructional Computing Support Team and a comprehensive Computer Demonstration Room. This paper describes how the room has come into being and the ways in which it is being used.

The data processing department of Wayne County Intermediate School District has sought since 1975 to meet both administrative and instructional needs of local school districts in the Detroit metropolitan area. Requests for our services have grown continually, and our resources now include an IBM 3033, 26 IBM 8100's and four Hewlett Packard 2000's. The IBM systems are used in a distributed network to provide administrative support. The HP's offer computerized guidance services through more than 60 terminals in schools and social service agencies. An additional 140 terminals access both the mainframe and the minicomputers to obtain multi-language programming resources and an extensive courseware library.

Over the last couple of years, we have been reexamining our facilities and services to determine how we might best respond to the growing use of microcomputers by Wayne County educators. We felt there were three alternatives:

--To view microcomputers as competitive systems, and to let educators who chose to use them fend for themselves:

--To accept microcomputers in general as a valid means to accomplish many instructional objectives, and to assist our locals as much as possible no matter what computing resources they chose to use;

--To endorse a particular microcomputer configuration as the preferred resource in certain computer applications, and to direct our support specifically toward that configuration.

We chose the second alternative. We felt either of the other choices would separate us from too many activities that were already taking place within local districts. To fail to recognize and support microcomputers would have been to ignore the many worthwhile ways microcomputers can and already are being used in county classrooms. To select only one system to support would have segregated us from schools that might have good reasons to prefer something else. It was clear that many educators in our county were going to use micros whether we supported them or not. It was also clear that many of them had strong ideas about which machines would serve their needs best. We saw little to gain by failing to recognize these realities.

Our next concern was to find a way to provide support on a level comparable to that of our other data processing services. This has proven to be a substantial undertaking. Potential microcomputer users need to become "computer literate;" they need to be able to compare and evaluate a variety of systems; they need assistance shopping for hardware and getting it set up and running properly: they need maintenance service; and they need to be able to locate, evaluate, and acquire appropriate software. Our goal at the Intermediate has been to provide all of these service for the two or three machines that have been most popular within our local districts while at the same time offering at least some degree of support for most other microcomputers being used in Wayne County classrooms.

The heart of our program consists of a four-member Instructional Computing Support Team and the Computer Demonstration Room the team directs. The team contains experts in hardware, software, and communications on systems of all sizes. In addition, the team has a combined total of over 40 years experience in computer-related education. While each member of the team has unique areas of expertise, most microcomputer-related activities are shared by the entire team.

Primary among these activities is the development and operation of the Computer Demonstration Room. This facility currently contains 30 microcomputers representing 10 manufacturers - APF, Apple, Atari, Commodore, Cromemco, Exidy, Heath, Hewlett Packard, Tandy, and Texas Instruments. CRT and printing terminals are also available for those wishing to explore our timesharing resources. An Advent projection TV may be used to display computer output to large groups. The room also contains a complete IBM 8100 system to train local districts' data processing staffs.

Getting the room into operation has been a major task for the Instructional Computing Support Team. Hardware selection and acquisition, furniture procurement, layout and wiring, and creating a secure environment have all been time-consuming activities that are never completely finished. We often feel it is almost impossible to keep up with new hardware and software developments, and we are finding that we sometimes have to sell as well as buy equipment in order to maintain a manageable inventory.

The team has also found that it has had to devote considerable attention to communicating among team members and between the team, the data processing department, and area educators. We have established regular hours for certain room functions, and have posted a calendar in data processing hallway to minimize scheduling conflicts. Our biggest problem will soon be finding time to handle the many requests for presentations that we are receiving from local districts.

As the room comes into its own we are finding that it enables the Intermediate to provide a variety of valuable, sometimes unique services. The Computer Demonstration Room:

--Offers educators an opportunity to compare and evaluate microcomputer systems, peripherals, and software;

--Provides a facility where microcomputers can be compared side by side with timesharing terminals and systems;

--Gives the Instructional Support Team a site to conduct training programs both for area educators and for the staff of the Intermediate;

--Enables the Instructional Computing Support Team to conduct research and to improve its expertise in instructional computing:

--Serves as a clearinghouse for the dissemination of public domain software to local educators and other ISD's;

--Is becoming a comprehensive center of microcomputer information as we acquire educationally relevant books, magazines, and promotional materials;

--Will gradually be opened to the general public so that county parents and children can benefit from exposure to the resources we have assembled.

Although we are pleased with our efforts thus far, we have found that the Demonstration Room has not yet fully lived up to its potential. The missing ingredient has been a sufficiently rich collection of software to answer the question, "What useful things can I do with a computer in my classroom?" Only a small minority of today's teachers have the time and expertise to develop their own instructional materials. For the rest, a computer without software is no more helpful than a movie projector without film.

We have taken several steps to remedy this problem:

--We have arranged to display and distribute software from MECC, the Minnesota Educational Computing Consortium. This exemplary instructional software for the Apple is available to area educators on 20 diskettes for $50 per machine;

--Through subscriptions to publications such as Creative Computing, The Computing Teacher, Purser's Software Review, and School Microware, we have identified most of the developers of educational software for microcomputers. We have written to all of them requesting that they make samples of their products available for display in the Demonstration Room. To date we have written nearly 150 letters and have received favorable responses from over 30 vendors, including many who offer the most significant products. We are going to follow up shortly with another letter to those who did not respond the first time;

--We are purchasing software that is not strictly educational but which we nevertheless feel it important to have. Utility programs, word processors, data managers, and programs that show off unique machine capabilities fall into this category. On occasion, we also purchase educational software that a vendor is unwilling to make available without charge;

--We are working with two neighboring Intermediate Districts, Oakland and Macomb, to acquire, catalog, review, and distribute useful public domain programs for several microcomputer systems. Current plans call for hiring parttime technical staff and summer professional help to get this project under way;

--We are discussing with our tri-county Professional Resource Centers arrangements for the PRC's to take over the acquisition and maintenance of most of our educational software. The team and the PRC's agree that there are many parallels between software and other educational media, and that the PRC's are better qualified to handle the organization of a large collection of programs and documentation. In addition, the PRC's have an on-going media loan program that will enable educators to examine software at their own buildings.

The Instructional Computing Support Team and the Computer Demonstration Room have both acquired significant popularity despite their limited publicity to date. The Demonstration Room was receiving 30-50 walk-in visitors each week even before we mailed an opening announcement. The team has been virtually inundated with requests to provide in-service workshops both at the Demonstration Room and at local district sites.

It is clear that we have tapped an important need in our educational community, and that serving that need will be a major challenge. We are proud that Wayne County Intermediate District has taken a leading role in providing these services to our community, and we are optimistic that our resources will continue to grow as local educators find the computer to be an increasingly indispensible instructional tool.

# THE MICROCOMPUTER AND MANAGEMENT
# of the
# TIME BOUND EDUCATIONAL PROGRAM

David K. Mosow
University of South Alabama
Mobile, Alabama 36688

The Teacher Corps Project at the University of South Alabama has developed a management system involving the microcomputer. The system utilizes a data base containing staff activities as they link to program objectives. Sorted by objective a report can be generated to help the project director determine which objectives are recieving adequate attention from the staff and which objectives fall short. Staff resources can then be redirected to better meet program goals.

The increasing access of the average American to microcomputers represents a significant change in the social process. It has been estimated that by 1985, nine out of ten Americans will possess a microcomputing device. Folk (1978) suggests there will be more microcomputers than automobiles per household in the United States in the late 1980's. Cornish, on the other hand offers a more conservative prediction (1981). He estimates that by 1990 over half of the American homes will have access to a microcomputer or own a terminal with telecommunications capability. The success of microcomputing is attributable to both its increasing accessibility and decreasing cost. Highly versatile units, allowing individualized use, are available for less than $1,000. This personalization of the microcomputer is the beginning of America's newest technological craze much like the car or the television.

## Implications

Microcomputing technology is especially significant for the field of schooling and education. The increased social acceptance and use of the computer among the general population will create a demand to develop computer literacy among young people and adults. Secondly, the microcomputer will become a practical and effective mechanism for improving the quality and efficiency of the instructional process. Computer assisted and computer managed instruction will gain increasing acceptance. As an example of this growing trend, Louis Frensel estimates that fifteen percent (15%) of all microcomputers sold in 1981 will be sold to educational concerns. These data processing capabilities are primarily available to students in the areas of science and math. It is estimated that thrity percent (30%) of the nations's high schools use computers, while only five percent (5%) of the elementary schools do. Such trends, with the easy access to simple microcomputer systems and the succession of low-cost technology, continue to grow and impact at all levels of education.

The Bell & Howell Corporation suggests through its marketing analysis that the sale of microcomputers to school districts will quadruple by 1982. They attribute these projections to the lower costs and greater accessibility of the new microcomputers. Those who are informed about computer technology suggest that the next great educational crisis will concern computer literacy. The rapid assimilation of microcomputer technology will require a population conversant and technically familiar with the computer. Donald Michael in the UNPREPARED SOCIETY (1968) suggests that ignorance of computers will render people as functionally illiterate as ignorance of basic communication skills. The future equity of educational opportunities depends upon equal access in obtaining basic computer literacy.

## Schools and Computers

Growing public demand and increasing dependency upon microcomputer technology will impact the educational arena. There will not only be a demand for training but for application of the computer as a school management and instructional tool. Computer assisted instruction is now generally acknowledged as one means of improving student learning. In this form the computer serves as a: (1) medium to individualized instruction, (2) a diagnostic tool, (3) an interactive medium for providing immediate feedback, and (4) a time-saving tool in constructing and providing problems. The use and application of the microcomputer seem bound only by the imagination. Perusing in the microcomputer literature, one notes that schools and other educational institutions have used these computers for keeping daily attendance, equipment inventory, textbook inventories, etc. The biggest problem in use of microcomputers is the lack of trained personnel to train others.

Those involved in the training of teachers are generally unfamiliar with microcomputer technology and consequently few computer-training requirements exist in preservice or inservice programs. Many educational institutions are not preparing programs to adapt students and/or teachers to this emerging technology.

## Project Applications

The University of South Alabama houses a federally funded Teacher Corps Project which is designed to provide inservice education to teachers dealing with troubled or delinquent youth. To achieve this goal, the Teacher Corps Project is committed to the use of microcomputers for instruction and management in order to improve the educational climate for the student. The thrust addressed in this paper is the project management information system and use of the microcomputer.

Being a federally funded program, it becomes time bound. That is, the funding will terminate at a predetermined time. To achieve the program goals during this predetermined time interval, a set of objectives was established. Two premises are put forth with respect to the program objectives:

1. The attainment of these objectives constitutes a successful program.

2. The objectives can be monitored by monitoring staff activity.

First, the premise is accepted that those objectives set down in the grant proposal document are important and, in fact, they operationally define project influence. Thus, meeting those objectives constitutes a successful program.

Secondly, the premise is accepted that staff members and their activity channelled toward the program objectives will contribute to the attainment of those objectives.

To monitor combinations of program objectives and relative staff activity, a data base management system was designed to run on an APPLE II plus microcomputer. The system hardware is configured to allow 48K bytes of internal memory. Auxiliary memory devices include two, five and one-quarter inch (5 1/4") disk drives. The output devices are an Anadex 9500 matrix quality printer and a small black and white monitor. The software utilized to drive this information system is "The Data Base Management System" by High Technology, Inc.

To build this data base, each staff member completes a small document as he/she performs an activity of significant magnitude to be classified as relating to one of the program objectives. On this data entry document (See Figure 1), the following information is captured:

1. Date.

2. Related Program Objective.

3. Activity.

4. Individual Staff Member Reporting.

5. Hours Spent.

6. Others Involved.

In an interactive mode, the information is keyed into disk memory. It can then be sorted and sub-sorted by any of the six categories. A sort by objectives, sub-sorted by date will give a chronological description of all activity related to each objective for a time period set by the individual generating the report (See Figure 2). By utilizing the summation of hours spent with respect to each objective, it becomes obvious which objectives are requiring most of the staff activity and which objectives should be investigated to determine if more concentrated effort is needed.

This information has allowed the Teacher Corps Project to make mid-course corrections in staff effort long before the program reaches a point where recovery would be extremely difficult.

## Spinoffs

After seeing the data base concept in operation, the University of South Alabama has purchased additional APPLE II's to track other segments of teacher education. The Mobile Public School System, Staff Development Center, is investigating the use of the APPLE II to track district employees and the staff development activities in which they have been involved. Several district administrators and local school officials are actively pursuing purchasing microcomputers for computer managed instruction, computer assisted instruction, and student records management. In addition, the State of Alabama Educational Information Center has adopted the data base concept to make educational information available to high school and junior college students state wide.

## References

Braun, Ludwig. "A Speculation on the Impact of LSI Technology on Computing" in *Computers and Communications-Implications for Education.* Academic Press, Inc., New York, 1977.

Braun, Ludwig. "Computers and Education--The Promise and Fullfillment" *APPLE*, Vol. 1, No. 1, 1979.

Braun, Ludwig. "What We Can Do With Micro-Computers That We Couldn't Do Before in Education," Unpublished Paper.

Elliott, Bob. "Programming in the Classroom," *Mathematics in School*, Vol. 8, No. 2, Pp. 11, March, 1979.

Folk, Mike. "Should Your School Get A Micro-Computer," *Mathematics Teacher*, Vol. 71, No. 7, October, 1978.

Frensel, Louis, *A Crash Course In Microcomputers*, Sams Publishing Company, Inc., Indianapolis, 1980.

Herstein, Ed. "Choosing A Classroom Microcomputer," *Computing Teacher*, Vol. 6, No. 4, Pp. 54-55, May, 1979.

McIsaac, Don. "Conquering the Fear of Flying with Computers: The State of the Art in Educational Management," an unpublished paper delivered at the University of South Alabama, June, 1979.

Michael, Donald N. *The Unprepared Society*, Basic Books, Inc., New York, 1968.

*Microcomputer Sourcebook for Educators*. Fort Worth, Texas: Radio Shack, 1979.

Milner, Stuart D. "Teaching Teachers About Computers: A Necessity for Education," *Phi Delta Kappan*, Vol. 61, No. 8, April, 1980.

Milner, Stuart and Carol Hargen. "Microcomputers...The Future is Now," *The Practitioner*, Vol. VI, No. 1, October, 1979.

Molnar, Andrew. "The Next Great Crisis in American Education: Computer Literacy," a paper presented at the meeting of the Society of Applied Learning Technology, Orlando, Florida, 1978.

Piele, Donald T. "Micros 'Go To' School," *Creative Computing*, Vol. 5, No. 9, Pp. 132-34, September, 1979.

Figure 1. Data Input Document

MOBILE TEACHER CORPS
MANAGEMENT SYSTEM

B 5.1.1.1

DATE: ___ ___ ___

OBJECTIVE: ___.___ or ___.___

ACTIVITY: ______________________________

INDIVIDUAL: ______________

HOURS SPENT: ___.___

OTHERS INVOLVED: ______________________________

SITE/TARGET: ______________________________

DOCUMENT CODE: B ___.___.___.___ or B ___.___.___.___

Figure 2. Sample Report

```
                                         QUARTERLY REPORT

DATE     OBJ         ACTIVITY                                                  HOURS

6/ 4/80  3.0   COUNCIL EXECUTIVE CALL MEETING                                  02.25
                 MANCE        WHITEURS TAYLOR SMITH

6/ 4/80  3.0   COUNCIL RETREAT PLANNING                                        01.25
                 MANCE        WHITEURS

6/ 5/80  3.0   COLLABORATION FOR SUMMER PROGRAM                                02.00
                 MANCE        SAM JONES

6/ 6/80  3.0   CETA SUPERVISORS MEETING                                        01.50
                 MANCE        LASANDRA LUDGOOD

6/ 6/80  3.0   SUMMER PROGRAM PLANNING                                         02.00
                 MANCE        JOE SMITH

6/ 6/80  3.0   INTERN WELCOME ABOAD PARTY                                      02.00
                 MANCE        TEACHER CORPS STAFF INTERNS OLD STAFF

6/ 6/80  3.0   SCREENING COMMITTEE FOR SUMMER PROGRAM STAFF                    00.75
                 MANCE        JOE SMITH PAUL WHITEURS CLARETA WITHERSPOON TOM HEWITT

6/ 6/80  3.0   COLLABORATION FOR SUMMER PROGRAM                                01.50
                 MANCE        SAM JONES PAUL WHITEURS NELVA SIMON

6/ 6/80  3.0   SUMMER PROGRAM PLANNING SECURING YMCA FACILITIES                01.00
                 MANCE        HENRY HELTON
```

# COMPUTERS IN EDUCATION - MEETING THE CHALLENGE

Robert W. Nelson
Milwaukee Public Schools
Washington High School
Milwaukee, WI 53210

Henry S. Kepner, Jr.
University of Wisconsin-Milwaukee
School of Education
Milwaukee, WI 53201

The Milwaukee Public Schools adopted the High Schools Unlimited Plan during the 1975-76 school year. This plan was used as the focus for developing the Computer Data Processing Career Specialty Program* at Washington High School. Substantial input and planning, drawing heavily on business, industry and educational leaders, has resulted in a high school career education program in computers and data processing, which is second to none. Currently, over 300 students from throughout the Milwaukee area attend Washington High School for this program. What follows is a discussion of the Computer Data Processing Career Specialty Program's background, major planning and implementation considerations, some indications of this program's success, and implications for future activities.

## BACKGROUND

The Milwaukee Public Schools embarked on a High Schools Unlimited Plan during the 1975-76 school year. Concurrently, the administration and staff at Washington High School, along with business representatives from the Milwaukee area, were exploring educational opportunities in the area of computer data processing. Judge Reynolds' decision, resulting in court ordered desegregation of Milwaukee Public Schools, served as a catalyst in the designation of Computer Data Processing as Washington High School's career specialty under the High Schools Unlimited Plan. Listed below are several highlights of the High Schools Unlimited Plan as adopted by the Milwaukee Board of School Directors. It is upon these concepts that the Computer Data Processing Career Specialty Program at Washington High School was developed.

"—From both a physical standpoint and a financial standpoint, a single high school cannot offer the variety of educational and career training skills required...in an urban society becoming increasingly specialized and technical.

—...each high school would offer a career specialty not available in any other school.

—Our high schools could offer these specialized programs more economically than having a single school attempting to offer a smattering of everything.

—Students must have free (transportation) access to any high school and any cooperating part of the community.

—...will require...the involvement of...business, labor, industry and any cooperating part of the community.

—...will foster a natural integration of students...because the schools will draw pupils city-wide."

A most important factor in the development of the Computer Data Processing Career Specialty Program at Washington High School was the formation of a Technical Advisory Committee. It was because of this group's initial support in the investigation, planning, design and staff training that we were able to begin our program during the 1976-77 school year.

---

*Jason Bukvich, 1979 winner of the AEDS programming contest, participated in this program from 1976 to 1979.

## MAJOR PLANNING AND IMPLEMENTATION CONSIDERATIONS

Each of the following components represents a significant element, critical to the success of the project. These elements are often interrelated, but are isolated for purposes of discussion.

### Technical Advisory Committee

The Technical Advisory Committee has representation from three areas. First, the business community—among them the most influential computer experts in the Milwaukee area. These individuals are managers/officers in some of the largest corporations in our area, as well as some of the smaller firms. Next, are representatives of post-secondary institutions. These individuals represent both technical schools and four-year colleges/universities. There are also key Milwaukee Public Schools staff members with responsibility for successful implementation of this program, including central administration representatives from Curriculum and Instruction, from the Data Processing Division and Washington High School's principal.

The role of the Technical Advisory Committee cannot be over-stated. This group meets as a whole every 5-6 weeks, and in subcommittees as needed. The Technical Advisory Committee takes a very active role in program design, development and evaluation. Each major decision affecting this program is reviewed by the Technical Advisory Committee, and, where appropriate, it makes recommendations. Through individual company efforts we have: (1) moved into a five-year planning cycle; (2) received enthusiastic support at each juncture from numerous areas related to program expansion and development; (3) developed a new small business computer component implemented in January 1980, which includes a small business computer system supporting vendor supplied application software; (4) continued to train "new" staff members at industrial sites; (5) received broad support from their professional organizations; (6) established field experiences and internships for our students; (7) received assistance with program promotion; (8) established one of the most successful of Milwaukee's career specialty programs.

### Student Access

Improving student accessibility to our program by eliminating inconveniences and unnecessary procedural barriers has been a key task along the way. The program has been designed to accommodate a wide range of student abilities and interests. Many students planning careers in computer data processing continue their education in technical school or college. The program continues to serve a number of students with exceptional educational needs. Students make commitments to the program one semester at a time and can participate for four full years. There is no transportation cost to students participating from outside Washington High School's district. Non-district students may attend Washington High School on a part-time basis if they desire to remain affiliated with their home school (about 1/3 of our students do).

### Program Staff

Currently, three departmental areas are represented directly in our program, including mathematics, business education and science. One strategy to develop and maintain up-to-date backgrounds for staff members has been on-site training at local businesses. Short-term employment is available from business and industry for appropriately trained staff. Instructors also attend/teach district inservice workshops and post-secondary courses. We are relying primarily on in-house training and support of committed teachers to accommodate program growth and expansion.

The equivalent of one staff position has been released to manage this program. This is split between a computer systems manager and a program implementor.

### School District Support

Support from the school district has been multifaceted and varied. The High Schools Unlimited Plan adopted by the Milwaukee Board of School Directors provided a structural basis for accepting new, non-traditional approaches in funding, student access, staffing, staff development and curriculum design and development. This, coupled with a high level of involvement by the private sector, has engendered, at worst, acceptance of the concept, and most frequently, enthusiastic support.

The central administration has been made directly accessible to specialty program directors (implementors). This has been most helpful in the areas of transportation, accounting, purchasing, budget, curriculum and instruction, relationships, instructional resources, duplication, pupil personnel, exceptional education, research, administrative services, and most important for our program -- data processing.

Also critical in supporting the magnet concept has been the district-wide support of groups, such as guidance directors and academic area department heads. Many inservice sessions for staff members are run through the Washington High School facilities. Providing staff and facilities for instruction to inservice programs has helped to promote Washington High School's specialty throughout the district. Currently, over 200 district personnel are attending computer-related workshops weekly.

The enthusiastic support and cooperation of our building principal has provided a measure of support which is absolutely essential. His role on our Technical Advisory Committee has been active and regular. The interface available to other district schools through the principal has been beneficial in countless situations.

When everyone works together toward a common goal, a seemingly insurmountable task becomes feasible. We are fortunate to be a part of this type of cooperative effort.

### University Support

Since 1972 Washington High School's mathematics program has been closely aligned with Milwaukee area teacher education programs—especially mathematics education from the University of Wisconsin-Milwaukee. Initially, there were high concentrations (as many as seven per semester) of university students doing their student teaching in the mathematics program. This has resulted in a particularly high degree of professionalism being displayed by staff members. The dialog with UW-M, which was prompted by the student teaching center, served as a catalyst for program staff to work together. Several regular staff members received their initial training via the student teaching center. A student teacher center for Wisconsin's newly instituted state certification in computer science has also been developed.

Also stemming from this relationship has been a series of college level inservice programs offered on-site. The development of the specialty program, beginning in 1975, provided an increase in the need to offer inservice training at Washington High School.

Having university personnel on-site regularly and frequently has been most helpful in program planning and design. Often, they have provided a needed incentive or forum for planning sessions. Although the pool of new people acquiring teacher training has diminished severely, UW-M's involvement has not diminished. In fact, co-author, Henry Kepner, has taught courses in this program for our specialty students.

## INDICATIONS OF PROGRAM SUCCESS

Although there are no absolute measures of program success available, there are numerous examples available which are encouraging to us and suggest we are heading in the right direction.

### Student Placement

From the outset our position has been to encourage a student continuing his/her education beyond high school. The Technical Advisory Committee said it would be an exception when a student would be hired into computer data processing directly from high school. Yet, many of our students are finding entry-level computer-related employment while continuing their education. Others have gone directly to work on a full-time career basis. Typical experiences have ranged from data entry-clerical positions to computer operations at places like Evinrude, Woolworth, Northwestern Mutual Life, Harnischfeger and Harley-Davidson. Students are also involved with software and systems development at the J.C. Penney Company, Advanced Technology Associates and R.J. Karek. Others are working as operations attendants at sites such as the Wisconsin Telephone Company and as programmers at Northwestern National Insurance.

### Business and Industry Support

Business and industry support continues to expand well beyond our Technical Advisory Committee. Included are computer-related professional organizations, such as the Data Processing Management Association. Several companies have standing openings for interns and graduates. The Technical Advisory Committee has maintained essentially the same nucleus of members during the past six years.

### Expanded Facility Usage

It is expensive to allow our computer resources to sit idle after the regular school day. Hence, an expanding number of groups are using our facilities as late as 9:45 p.m. weekly. Among these are the regular day time students. Adult groups include community education, district staff inservice and university classes. Beginning in the fall of 1981-82 there will also be evening high school offerings.

Another use is emerging. Visitors are finding our staff, program and resources a convenient forum for planning discussions. This mutually beneficial activity serves to keep us abreast of related efforts at local, regional, national and international levels.

### Program Growth

Our regular student population has grown from 144 students in our first year (1976-77) to 307 currently. This growth has been accompanied by a parallel growth in program resources. Included among the students are children of Milwaukee Public Schools staff members, exchange students from suburban districts and students from each of Milwaukee's 13 high school districts. The program has always been racially balanced. We have been able to serve a variety of exceptional education students, including deaf, physically handicapped, learning disabled, behaviorally disabled, emotionally disturbed and trainable mentally retarded.

New areas are emerging where we have become involved as a program. One notable area is in the use of microcomputers in 9th grade classes. We are beginning to systematically address computer literacy needs for a broader cross-section of our school's population.

## CONCLUSION

The challenge of building a high school computer program and facility that meets many needs has been presented. Washington High School has the computer data processing specialty program in the Milwaukee area. This specialty attracts students from throughout the metropolitan area. Several challenges remain for the staff at Washington High School.

First, the program to make all students at Washington computer literate must be carried to completion. A lab for non-specialty students and the implementation of computer literacy units in the 9th grade are moving forward, but the development of instructional software for use in courses throughout the building is a slow process. One approach has been teaming an advanced programming student with a teacher seeking to incorporate the use of a computer, but, again, the pace is slow. Staff development time is a major factor in this effort.

Second, is the challenge of keeping the Washington specialty program unique. With rapid technological development and seemingly lower prices for equipment, the effort to keep equipment, program and staff ahead of other school programs is increasingly important.

Third, the challenge of marketing Washington High School as a community resource is critical to the neighborhood. The current uses of the computer facilities by students, district inservice programs, and university instructional programs is scheduled for expansion. This expansion is a key component in providing a recognized, balanced school setting as a factor in preserving neighborhood stability.

The continued development of this unique program depends on the strong support and cooperative efforts of several groups. The Technical Advisory Committee, school staff and district administration must interact successfully in the development, revision and implementation of long-range plans. Community involvement and university support are necessary factors in making these plans a reality.

# ROBOT PROBE

## AN INTRODUCTION TO PROGRAMMING

Dr. Robert D. Postman
Mercy College
Dobbs Ferry, New York 10522

## ABSTRACT

Microcomputers are being introduced into the schools rapidly. We need a computing curriculum which develops the structural and practical elements of programming prior to any formal programming. Robot Probe was developed as part of such a curriculum.

Robot Probe uses a pseudo-language to introduce students to programming. A robot, probes and a space module are represented graphically. The student learns to enter commands which control the robot's movements. Students use Robot Probe in four phases (1) Single Commands (2) Programming (3) Editing (4) Creating Sub-Commands. Robot Probe is written in BASIC for the TRS-80 and requires 16K of memory. The program has been used successfully in grades three through six.

### INTRODUCTION

Uncounted elementary school students are being introduced to microcomputers and programming each year. This curricular innovation is occuring rapidly, perhaps more rapidly than any other widespread curricular change in our memory. We find ourselves unprepared to offer students a structured curriculum for learning about computing and programming.

Too often, a student's only exposure to microcomputers consists of some games of micro-baseball or some programs designed to teach mathematics or reading. If the student is fortunate, the teacher will know something about programming and will pass that knowledge along. While some exposure to programming is preferable, neither approach is adequate.

We need to offer students a well planned, interesting curriculum which develops the structural and practical elements of programming prior to the introduction of formal programming languages. Furthermore, such a curriculum must include computer programs which can be used in the classroom. That is, these programs should require a minimum of memory and other support. Robot Probe is such a program.

### OVERVIEW

Robot Probe uses a pseudo-language to introduce students to programming. In Robot Probe a robot "lands" on a planet along with four probes and a space m [illegible] module. The robot's mission is to pick up the four probes, load them into the module and blast off.

The robot, the probes and the module are represented graphically on the monitor (radar screen to students) with a grid design. The robot always appears at the same grid position. The probes and the module are randomly placed. The robot's movements are controlled by entering appropriate commands through the keyboard. These movements are shown through animation on the monitor. Each command is cued and the controller is given warning messages when the robot cannot execute or understand the command.

Robot Probe is written in BASIC for the TRS-80 and requires 16K of memory. Robot Probe is currently being adapted for use on the Apple and other microcomputers. Please refer to the last section of this report for a summary of Robot Probe Commands.

## USING ROBOT PROBE IN THE CLASSROOM

Robot Probe has been used with students in the third through the eighth grade and has proven particularly effective in the third through the sixth grade. However, older students and even adults who have no experience with programming have found Robot Probe a valuable introduction to fundamental programming concepts.

Typically, a Robot Probe "unit" lasts from two to five weeks, from three to five days each week. The teacher usually introduces Robot Probe to the entire class and then sets aside time during the day for individuals or groups of students to work on the program. Skill levels range from entering a single command correctly through finding the most compact program for a particular task.

Children go through four distinct phases when using Robot Probe:

1-Entering Single Commands
2-Programming
3-Editing
4-Creating Sub-Commands

### Entering Single Commands

When first using Robot Probe, students enter the commands singly and observe the effects of these commands on the monitor. At first, students use only the TURN and MOVE commands which orient the robot and move the robot horizontally and vertically between grid positions. Next the students learn the PICK and PUT commands which cause the robot to pick up and put down the probes.

Once these commands have been mastered the students take the robot through the sequence of commands necessary to load the four probes into the space module and then enter the BLAST command to blast off the planet.

While entering commands singly, the student can easily correct any mistakes and respond to any warning messages issued by the robot.

### Programming

Once the individual commands are mastered the student switches to the PROGRAM mode. This mode enables the student to enter an unlimited number of commands, line by line. The commands are not executed as entered but rather the entire sequence of commands is executed once the student types END as a command in the program. Each line of the program is prompted. Students begin with simple programs and progress to programs consisting of the entire sequence of command leading to BLAST off.

Obviously, more care is required for programming than when the commands are entered singly. The student cannot correct mistakes or respond to warning messages immediately but, rather, must wait until the entire program has been executed. Students find this activity challenging and interesting. It is an excellent simulation of the actual process of programming.

### Listing/Editing

As students work through the process of programming, the need for a listing and editing capability becomes immediately obvious. Students ask for a way to view the program prior to execution, and to change individual commands. Entering LIST as a command in the PROGRAMMING mode lists each line of the program. The EDIT command enables the student to change any line in the program. After some experience with the LIST and EDIT commands, students can produce error free programs.

### Creating Subcommands

A student can create subcommands made up of existing commands and subcommands. To create these subcommands, the student first enters SYSTEM and then SUB. Each subcommand is limited to nine component commands. Subcommands which "tell" the robot to TURN AROUND, PICK UP BOTH, and PUT BOTH DOWN are most popular. Students also create a number of other interesting, and often complicated, subcommands.

### The Next Step

Robot Probe has been designed to embody the structural and practical elements of programming. Students who complete this program have successfully moved on to programming in BASIC without some of the problems we have noted in students who start out immediately with the BASIC language.

Robot Probe does not have the branching and decision making components found in programming languages. For this reason, a second program designed to follow Robot Probe is being developed. Robot Probe II will contain these branching and decision making components.

## COMMANDS

### TURN (LEFT) (RIGHT)

Turn the robot 1/4 trun in the direction indicated. (Robots orientation!) (Shown by the arrow on the screen)

### MOVE (LEFT) (RIGHT) (UP) (DOWN) (numeral optional)

Moves the robot in the direction shown. (Controllers orientation!)

If not followed by a numeral moves one space otherwise moves the number of spaces indicated.

The robot must be facing in a direction before it can move in that direction.

### PICK (LEFT) (RIGHT)

Picks up probe on the side indicated (Controllers orientation.)

### PUT (LEFT) (RIGHT)

Puts down probe on side indicated.. (Controllers orientation.)

### BLAST

Blasts the robot and its cargo of probes off the planet. This command will execute only when the loading sequence has been completed successfully.

## PROGRAM MODE

This command switches the controllers machine to program mode. The controller can enter an unlimited sequence of commands, line by line. Once a program is executed it is erased from memory.

### END

Entering END as a line in program mode terminates the program mode and executes the program.

### LIST

Entering LIST as a line in the program mode lists all the lines, by number, one at a time and prompts the next line entry.

### EDIT

Entering EDIT as a line in the program mode lets the controller change lines in the existing program.

## SYSTEM

This command gives access to two other commands, SUB AND GOTO.

### SUB

The SUB command permits the controller to create new commands made up of existing commands and subcommands. There is a limit of 10 subcommands and each subcommand is limited to 9 component commands.

### STOP

Entering STOP as a component command fixes the subcommand and returns the machine to the command mode.

## OTHER COMMANDS

### LAND

Changes position of the robot and the probes on the radar screen.

### RESTART

Returns the robot and probes to their original starting position.

## ERROR MESSAGES

The robot issues these error messages.

"I do not understand your command"
"Can't move. Face me in that direction"
"That command would take me off the screen"
"No probe to pick up on that side"
"No probe to put down on that side"
"Not ready for BLAST OFF"

Every command is either executed or receives an error message.

# Management Operations Review and Evaluation (MORE) of a School District Computer Services Department

S. M. Raucher, and T. J. Koehler
Computer Services Department
Montgomery County Public Schools
Rockville, Maryland 20850

ABSTRACT: In this time of inflationary pressures on budgets and taxpayer militancy relative to the high cost of public services, it is imperative that every facet of school district support services be critically examined. Such a review should stress areas for improved efficiency, reduction of wasteful procedures and practices and optimized management to facilitate getting the most service possible for each tax dollar spent.

The Board of Education of Montgomery County (Md) mandated a study of each functional unit of the school district. The study of the Department of Management Information and Computer Services was one of the first completed. This paper summarizes the methodology used to evaluate the department. It includes management and organizational issues, M.I.S. issues, conceptual philosophy alternatives for providing computer services, operational and capacity issues and the future role of computers in public school education. The methodology (rather than the specifics) is expounded, together with generalized findings that will be applicable to most large data processing operations. Conclusions and recommendations, such as the need for long-range planning, distributed processing, and cost benefit are presented.

The Montgomery County Public School System is a large suburban school district situated northwest of and adjacent to Washington, D.C. The district services approximately 100,000 students attending 186 schools. It employs 11,600 professional and support staff at an annual cost in excess of $310,000,000. The district, although funded by the County Government is responsible to a seven member elected Board of Education.

As part of its commitment to the electorate, the Board of Education recently mandated a series of management audits for all functional areas of the school system except direct instructional services to students. These were called Management Operations Review and Evaluation studies and the acronym "MORE" was applied. The objectives of each of the 21 separate MORE studies were to determine answers to the following questions:

- Can any functions or services (of the unit studied) be eliminated?
- Can any functions or services be provided in more effective or more efficient manner?
- Are there ways to assume additional functions or provide additional services without adding new resources or in an otherwise cost effective manner?
- Is the organizational placement of the unit appropriate and effective in supporting the overall objectives of MCPS?

Before a specific MORE study could be conducted, however, these broad objectives were redefined as specific research and evaluation issues applicable to the unit being studied. While the issues for each study were not restricted to those which address the five overall objectives, the identified issues must enable the researcher to comment on the MORE objectives in the final study report, along with additional topics as appropriate.

The data processing functions of MCPS -- including (a) management of the functions, (b) systems development, and (c) computer operations constituted one of the 21 MORE studies. This study, which focused primarily on The Department of Management Information & Computer Services (DMICS) as the operational unit addressed the computer support of both instructional and administrative applications and was identified by both the Board of Education and the Superintendent as one of the highest priorities in the MORE schedule. The study was mandated to be designed and conducted under external contract to ensure the highest possible level of both technical expertise and objectivity.

Background of Data Processing in MCPS. In the late 1940's MCPS began using unit record equipment to assist in the processing of payroll and other financial functions. The first computer, an IBM 1401, was installed in the early 1960's and was used until an IBM 360/40 was installed in 1968. Following various upgrades to the 360, an IBM 370/145 was installed in 1971. From 1968 to 1974, MCPS also conducted research in Computer Assisted Instruction (CAI) with a 32-terminal IBM 1500 Instructional System. In 1974, the instructional uses of the computer were combined with administrative systems, and an upgraded IBM 370/158 was installed. Since 1974, the IBM370/158 and its performance have been continually upgraded to the present 3.5 megabyte OS/VS1 system. In addition, MCPS currently has more than 80 microcomputers in secondary schools, for problem solving and computer literacy education, and 200 CRT's used for administrative and instructional purposes in approximately 60 schools and the central office.

The growth of computer hardware has been accompanied by a growth in technical personnel, and staff resources which reached a peak of 102 positions in fiscal year 1975. Since that time, however, a gradual reduction of staff has occurred to meet fiscal constraints on the school system. The 1980-81 personnel allocation for DMICS is 95 positions; the operating budget is 3.1 million dollars.

During the early 1950's, a significant portion of the MCPS developmental resources were allocated to converting the earlier financial applications and developing the initial pupil systems. Personnel and financial applications received the majority of resources during the late 1960's and 1970's. Only in the past few years have pupil-related applications come to the fore. Instructional computing and computer-managed instruction continue to consume a major share of the DMICS service functions.

The MORE study of DMICS began in December 1979, and culminated with a 900 + page report and presentation to the Board of Education in January 1981. The total cost of the study (exclusive of MCPS staff time) exceeded $80,000.

### Methodology

During the first quarter of 1980, two contractors were selected to design the study and identify issues. The results of the design phase combined with the guidance and direction of MCPS staff were developed into the overall MORE study plan. The scope and methodology of the study were as follows:

1. The study was to develop objectives for The Department of Management Information & Computer Services (DMICS)

   These objectives were to define the role of DMICS in the organization, and in meeting the overall goals of MCPS. They were to be measurable, achievable within current and known future technology, and reflect the proper organizational position and internal structure of DMICS within the school system.

2. The Study was to Recommend a Requirements Priorization Planning Process

   A workable strategic and tactical planning process was to be developed. The planning process was to include:

   - Development of procedures and instruments for collecting computer user requirements
   - Prescription of the analysis necessary for sizing the impact of implementing projects on both the systems development staff and the future computer capacity requirements
   - Proposing approval methods for project prioritization, based on mandates (necessity) and cost-benefits (option)
   - Methods for translating computer systems requirements into budget items, for both head count (staff) and equipment
   - User input required for escalation of unfavorable priority decisions
   - A mechanism for tracking progress of in-plan items, including comparison of estimated vs. actual cost and time.
   - Methods for validating benefits after implementation
   - Timing for repeat of the planning cycle
   - Recommended organizational aspects of the planning functions, identifying who is responsible for identifying requirements, sizing the impact, justifying the need, prioritizing the list, approving the effort, budgeting the cost, implementing the result, and tracking the process.

   The planning process was to be tested in the MORE Study by gathering current requirements and building a capacity plan for use in the next budget cycle.

3. The Study was to Recommend a Long-Range Conceptual Computer Systems Strategy

   The long-range computer systems requirements which were gathered were to be compared with the functional capabilities of system-wide computer architectural alternatives available within current technology and expected trends. Only proven concepts were to be considered, recognizing the risks of being too innovative with an emerging technology. The costs in time and disruption of moving to the recommended implementation were to be analyzed and weighed against long-range benefits.

### 4. The Study was to Validate Current Computer Workload Analysis

All elements of the current computer workload were to be quantified, weighed against their value, compared against norms in the case of such elements as test time and reruns, analyzed as to logical period (shift) of execution and impact on the system and user community. The impact of possible incremental improvements to the configuration were to be assessed.

### 5. The Study was to Propose a Five-Year DP Capacity Plan, Based on the Recommended Systems Strategy

User requirements and expected growth factors were to be gathered, sized, and projected onto the current workload. Two projections were to be made: (1) for all identified requirements, and (2) those likely to be approved. The forecast was to be quarterly for the first two years, then annually thereafter. Projected aggregate workload was to be converted into system hours by shift and percent utilization for multiple compatible processor alternatives.

### 6. The Study was to Develop a Format and Methodology for Tracking Forecasted Requirements vs. Actual System Usage

The forecast model was to be calibrated by taking May 1979 data, (job accounting tapes) making a 2-month forecast, and comparing with actual July 1979 measurements, taking into consideration seasonal variability. MCPS-DMICS personnel were to observe the process of forecasting and tracking, in order to be able to repeat the process in the future using the methodologies provided.

### 7. The Study Recommended Management Information System (MIS) Strategy

A long-range strategy meeting the management-by-information needs of the organization was to be addressed, to complement the conceptual systems strategy. Its overall goal was to improve the productivity and manageability of MCPS by translating data into information for day-to-day decisions.

### 8. The Study was to Contain a User Satisfaction Survey

The overall satisfaction of the user community with current central-site and user-based micro-computer services was to be measured, user-identified problems and suggestions were solicited, and were used to point to areas where improvement was indicated. A separate user requirements survey was made to support the capacity planning activity, as described earlier. The survey can be repeated at regular intervals to track improvements.

### 9. The Study was to Develop a Strategy for Improving Teamwork Between DMICS and Users

The operational interfaces between DMICS and the user community, including current organizational decision flow, communications, and day-to-day problem-solving processes were to be reviewed to identify team-work building improvements.

### 10. The Study was to Conduct a DP Employee Attitude Survey

Maintaining a staff of competent, highly trained professionals is necessary to meet the DP service requirements of MCPS. Recruitment and retention of a qualified team requires careful attention to salary, benefits, and working conditions, since such skills are highly marketable in the Washington area. Changes conducive to maintaining on board critical skills were to be identified via a survey of the subject personnel, which can be repeated periodically to track problem areas and measure improvement.

### 11. The Study was to Develop a Physical and Environmental Requirements Statement Reflecting the Recommended Systems Strategy

A 1979 study of computer hardware physical and environmental space alternatives was to be amended to reflect changes in equipment and operations dictated by the recommended conceptual strategy, if any.

### 12. The Study was to Perform, an In-Depth Post-Implementation Audit of New Applications, Plus Audit Results for Selected Applications

"Those who do not learn from the past are doomed to repeat it." The application development process was to be audited by review, comparing estimates vs. actual resources required, evaluating tradeoffs made in the light of the quality of the resulting product, and judging the merits of the development methodologies used. Every significant future effort was to be subjected to such a review. The method combined the "Discovery" procedure of tracking events and communications (used by the legal community to document the chronological course of events in a case) with systems analysis and evaluation. This process would allow the implementers to weigh design practices and the logic of tradeoffs made, to point to future enhancements of the development process. It can also be used to get failing projects back on track, by backing up to the point where incorrect design alternatives were taken.

### 13. The Study was to Propose an Application-by-Application List of Recommended Changes to Operational Strategies and Procedures or Reimplementations

The cost of rewrite of some applications was to be justified solely on the basis of resources saved in production and maintenance over a short period of time.

When the study began, the Department of Management Information and Computer Services was faced with overwhelmingly demands for a "free" unlimited service which violated every law of queueing theory. Imagine the chaos at an airline check-in counter if all flights were free....and if everyone knew it. Therefore, attention to development of a requirements process, and the necessary organization enhancements and resource planning to complement it, were a major portion of the proposed study.

In conducting the study a "systems approach" was used. Data collection on some activities overlapped analysis of some other unrelated activities, and results of activities were documented in-line in essentially final form, so that interim results were available for early review. Provision was made for "technology transfer" of the methodologies used, which could become the basis for periodic internal repetition of study activities.

## RESULTS OF THE MORE STUDY

The completed study was documented in three volumes. These were: the Management Report, describing the major findings and recommendations requiring immediate management attention; the Technical Report, provides the detail and the study methodology; and Issues and Answers, which provides the responses to specific MCPS questions (249 of them).

Highlights of the recommendations of the study were:

- MCPS must establish a long-term plan and strategy to deliver the information services necessary to manage the organization effectively; i.e., a Management Information System (MIS) plan.

  The planning should start now, with task-force priority.

- MCPS needs a top-level Data Processing (DP)-MIS Steering Committee to administer the plan. This committee should be appointed to implement the MIS plan as soon as it is approved.

- If the MIS task force were established by January 1981, its results could be available in time for FY83 Budget negotiations. This is key to the implementation of a new conceptual system strategy by September 1985.

- The conceptual system should be one capable of supporting the school and central offices alike with the most economic and responsive services possible with then current technology, i.e. 1985

- The system strategy should be based on distributed-system architecture (distributed data and distributed function): a network of intelligent terminals in the schools for local school administration and instruction coordinated from a central network node in the Department of Management Information and Computer Services (DMICS) which would provide data security and backup, centralized reporting, and consolidated data for on-line, system-wide management information requests.

- The role of DMICS in the organization must be clarified and its scope of vision improved by placing its director on the administrative team and/or other high-level deliberative bodies, so that DMICS can be consulted concerning the effects of decisions on their ability to carry out the DP plan.

- Until the new system can be implemented, significant improvements must be made quickly to upgrade current hardware capacity, possibly in increments, in order to allow it to provide the necessary level of services with acceptable quality, between now and 1985.

- A new facility must be provided to house the central computer resource adequately in order to protect the MCPS investment in this equipment.

- The programming staff must be augmented to supply development services (1) for all mandated requirements which cannot be complied with by manual methods, and (2) for all requirements that can be cost/justified to the satisfaction of the Steering Committee with a pay-back period of three years or less as long as those requirements agree in principle with the long-range plan. This programming staff increase, plus the addition of a planning unit are the only major organizational and/or personnel changes recommended within DMICS.

The report acknowledged fully that the implementation of the recommendations represented millions of dollars in the school system budget for the next four years, but defended them in their entirety on the following ground: The report contended that the MCPS was paying currently for cost/justified appalications not implemented in recent years — paying in budget dollars for high administrative costs, paying with sometimes mediocre decisions because adequate information was not available to point to the superior decision, paying in terms of management frustration and lack of productivity. The report postulated that DP Costs are an Investment Not An Expenditure.

The report held that the MCPS has too much data, and not enough information. Too many mandated reports and not enough scribes. Too many outside queries and too many file cabinets to search for the answers.

Based in part upon the MORE study results, several significant actions have begun. Specifically, a capital project to build a new physical facility for data processing has been approved; operating budget funds for interim upgrades of the current computer configuration have been approved as have increased funding for training of DP staff. In addition, a high level task force has been commissioned to define the role of computer technology for the next ten years. The goals and objectives of this task force are as follows:

Goals

1. Establish a long-term plan (10 years) and strategy to deliver the information services necessary to manage MCPS effectively.
2. Provide a conceptual systems strategy capable of supporting economic and responsive services possible with the technology to be available in 1985/1990.
3. Define and implement procedures for monitoring the use of technology, setting priorities and managing the information system in a cost-effective manner.

Objectives

1. Identify the computer related requirements of the MCPS organization at all levels and in all functions including instructional, administrative and management uses of technology.
    a. Identify information central to the operation of MCPS, which requires full or substantial systemwide integration.
    b. Identify information that must be uniformly defined and available throughout MCPS, regardless of where the data originate or are produced.
    c. Identify information which is wholly or substantially of interest only to individual offices, schools, or programs and does not require systemwide coodination or integration.
2. Identify the current means of collecting, storing, and making information available to users.
3. Identify the extent and manner in which identified requirements are currently being supported by manual or data processing systems.
4. Identify additional or replacement systems currently planned or under development and evaluate these against identified requirements.
5. Recommend a course of action for information systems and data. Recommendations should include:
    a. A framework for the information systems and data files, each part representing a logical "project" for development purposes.
    b. Priorities addressing identified requirements and deficiencies, assigned by "project".
6. Recommend a course of action for implementing the proposed systems plan. Recommendations should include:
    a. Policy, operating, and control responsibilities for the segments of the plan;
    b. An implementation timetable.
7. Provide multiple options to reflect optimal, medial, and minimum resource levels necessary to implement the MIS and the impact of each level on the benefit to be derived.

Clearly the results of the MORE study and the projected results of the Task Force have and will provide increased visability and support for computer services. The MORE study has once again demonstrated that more investment in DP means more service for the school system and more effective use of tax dollars for everyone.

# THE ORGANIZATION OF USER SERVICES FOR INSTRUCTIONAL COMPUTING

Don G. Rawitsch
Manager of User Services
Minnesota Educational Computing Consortium
MECC, 2520 Broadway Drive, St. Paul, MN 55113

ABSTRACT:
The implementation of instructional computing in schools and colleges requires more than acquiring hardware and software. Training, information dissemination, and consulting support is also needed to provide the "people connection" between technology and educators. This is the role of user services. This paper will discuss many aspects of providing user services for instructional computing on a large scale basis including the following: topics and activities in a user services effort; staff roles and background needed for them; a model for staff organization and user coverage; a case for centralized user services; a description of a training network; and the evolution of the user services role.

## Introduction

The use of computers in the classroom is growing by leaps and bounds. This growth has been greatly accelerated during the past three years by the advent of affordable microcomputers. While educators might engage in lengthy debates over what kind of computing hardware to purchase, once the purchase has been made they soon realize that a more difficult problem lies ahead. People support must be provided to bridge the gap between technological capabilities and their usefulness to teachers. The area of training for educators in the operation of computers, their instructional applicatons, and their integration into the curriculum shall be refered to as "user services". This overview will discuss a number of aspects of providing user services for instructional computing on a large scale basis and will draw on the experience of the Minnesota Educational Computing Consortium (MECC) for specific examples.*

## Topics of User Services Activity

The areas of support needed by educators in instructional computing can be categorized as shown below:

1) training on the operation of computer equipment in the classroom - Teachers need to know how equipment is installed, operated, and maintained.

2) training on the operation of applications programs on a classroom computer - teachers need to know how to instruct and provide responses to the various applications available on computers.

3) training on the use of applications in the instructional process - Teachers need to know how various computer applications can fit into the instructional sequences they use to cover various areas of learning and how computer related materials can be combined with non-computer materials.

4) training on the creation of individual computer applications - Teachers will find that not all of their own needs for computer related materials will have been created by others and they will want to create materials that fit their individual classroom situations.

---

*Some of the specific information in this paper is drawn from the author's work in preparing "A Review and Analysis of User Services Support For Instructional Computing In Minnesota: A Report To The Minnesota Legislature", published by MECC and the Minnesota State Department of Education, St. Paul, January, 1981.

5) design and development of appropriate computing courseware - Educators with instructional design skills and subject area knowledge must be given the opportunity to create computer related applications which can then be programmed for the computer by people with the appropriate technical skills since most teachers will not have the background, resources, or motivation to create their own applications.

6) coordination of standardization of materials and methodology - Computer hardware and software can be changed easily by users themselves and thus some organization is needed to review suggestions and then set standards regarding the purchase of products, design of materials, and usage practices so that efforts are not duplicated and materials produced are applicable to many.

7) advice on acquisition of computing products- Users will become aware of products and services available in the areas of equipment, maintenance, software, and documentation, and they will want to seek knowledgeable advice before spending public money.

8) facilitation of sharing instructional applications among users - Users who do create their own applications will want to share them with others and in return receive the results of others' work, requiring the availability of a clearinghouse arrangement.

9) information on adjusting to technological changes - When commercial vendors change products or introduce new ones educators will want to know how or whether these changes will affect their efforts and if it is advisable to take action.

10) information on the impact of computers on society - Instructional computing is not only learning to use computers but also learning about them, and educators will need to find information on the social impact of computers to convey to their students.

Types of User Services Activity

The areas of support listed above are served through a variety of activities. People engaged in full time user services work will find themselves spending their time on the following:

1) visitations - Educators feel most comfortable when they can learn about something new in their own familiar environment. Periodic visits to sites being served allow the services person to become familiar with the computing environment first hand, to answer user questions face to face, and most importantly to build trust within the users. Travel is inherent in user services work.

2) informational presentations - The world of computing changes rapidly and users are always eager to keep up with it. Beginning users especially have a need to be brought up to date with information about what services are available and how they can be obtained. Informational presentations to groups of teachers, parents, school administrators, school boards, and professional groups are a significant part of user services.

3) training workshops and classes - Beyond receiving background information, educators want to acquire specific knowledge and skills that will help them better utilize the computer in their work. An important part of user services work is teaching, either through single session workshops or formalized classes. This type of activity also requires that formalized training materials be developed.

4) meetings for local computing coordinators - Each site receiving services probably has a key person assigned to coordinate instructional computing activity there. User services staff not only provide help to these people, but also expect their advice and cooperative effort in return. Special meetings should be held periodically with these key facilitators so that they have input to larger scale planning efforts.

5) newsletters - Some kinds of information is best distributed in writing. User services staff will need to publish newsletters of some kind.

6) courseware development - The availability of quality courseware is critical for the success of instructional computing. In a large operation such as MECC's, responsibility for planning and coordinating this development should be handled by a staff group separate from the training focus of user services. However, user services staff will undoubtedly be utilized to accomplish some of the development work in the courseware area. Besides the design, programming, and evaluation work that might be assigned, user services staff, because of their constant contact with school users, are valuable for finding out which applications are most needed and for lining up classroom environments for field testing new courseware materials.

In addition to the above major activity types, user services staff might also become involved in projects such as the following:

organizing computing conferences
organizing computing contests for students and teachers

assisting users in the preparation of local grant proposals
conducting tours of local computer center facilities
participating in national professional conferences and submitting articles to local and national publications.

Finally, it is natural for user services staff to be perpetually involved in daily consulting with users, via telephone and (in MECC's case) electronic mail.

Attributes of User Services Staff

The person who is going to cover the topics and engage in the activities listed in the previous two sections must be multi-talented. Of course, in selecting someone to work on a user services staff, one would look for attributes typically associated with "good" employees: hard working, helpful, pleasant, and so on. The characteristics needed specifically for this type of work are listed below:

Skills -

1) teaching skills - The large amount of time spent making presentations and teaching workshops and classes demands that the person be expert in organizing, sequencing, and presenting material in a way that helps people understand and remember it.

2) programming skills - A user services staff person needn't be a professional programmer but will need skills in this area for helping users with programming problems, teaching sessions on how to program computers, and creating new applications for classroom computing.

3) curriculum design skills - The creation of new applications and construction of workshop and other training materials requires a knowledge of how to design and sequence learning activities, often in written form, so that teachers find them valuable for classroom use.

4) writing skills - User services staff are called on to do a lot of writing for design of curriculum materials, production of newsletter articles, and correspondance with users. Skills in written communication are highly important for doing these things well.

5) problem solving skills - This type of service might require being "on call" for up to fifty key "customers" and they will never run out of problems for which they need help. Being able to reassure dissatisfied users, plan ways to solve their problems, and do this for many people simultaneously is an indispensible skill for user services staff.

Personal Attributes -

6) creativity - There are always new ways to do things which might be more effective or interesting than the old ways. Whether it is design of materials, teaching a class, or solving problems, creativity will make a person more effective in serving users.

7) initiative - Computing is such a new field for so many educators that people supporting instructional computing will never see the end of requests for assistance. A service person must know how to "run his own show" much of the time and respond to user problems promptly.

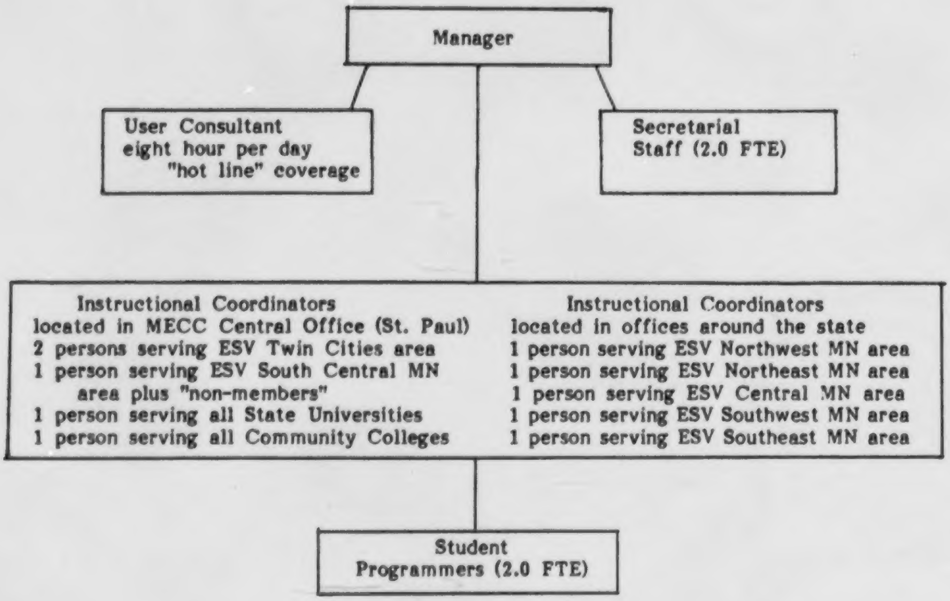
8) diplomacy - In any newly emerging field, different people will have varying ideas on how things should be done. The user services person will probably work for an organization with policies of its own. He must be able to work with people within those policies while making them feel that flexibility exists to meet their needs as often as possible.

9) background - Even a person with all of the above characteristics will have his credibility partially judged by his previous experience. Most valuable will be a background that matches that of those being served. Thus, a person providing user services to elementary and secondary schools will benefit from having once been such a teacher in terms of knowing the demands and politics of that work situation. A user services team would benefit from a mix of backgrounds, including people from different subject area specialities and different "grade" levels (elementary, secondary, and college).

## Model For Organization

User services on greater than a localized scale needs a model of organization. Ideally this model should balance the ability to work with individual users on a personalized basis with the need to team up staff internally for certain projects and promote a team morale. The MECC model for serving the state of Minnesota contains the following components as diagrammed below:

1) management
2) direct service to specified user constituencies
3) secretarial and programmer assistance.


Manager
User Consultant
eight hour per day
"hot line" coverage
Secretarial
Staff (2.0 FTE)
Instructional Coordinators
located in MECC Central Office (St. Paul)
2 persons serving ESV Twin Cities area
1 person serving ESV South Central MN area plus "non-members"
1 person serving all State Universities
1 person serving all Community Colleges
Instructional Coordinators
located in offices around the state
1 person serving ESV Northwest MN area
1 person serving ESV Northeast MN area
1 person serving ESV Central MN area
1 person serving ESV Southwest MN area
1 person serving ESV Southeast MN area
Student
Programmers (2.0 FTE)

Notes: FTE = full time equivalent staff
ESV = elementary/secondary/vocational school coverage
"non-members" refers to users not formally a part of the MECC consortium, such as private schools and state agencies

A MECC Instructional Coordinator must be able to constantly allocate time among the various activities listed in the earlier section of this paper. Below is shown a rough approximation of this allocation and some statistics on activity carried out by Instructional Coordinators in the MECC User Services Department, who over the years have numbered about ten.

| | 1979-80 | 1974-80 |
|---|---|---|
| Site Visitations (20%) | 507 | 3572 |
| (Note: The 507 visits last year covered 93% of the roughly 370 school districts and colleges in Minnesota using instructional computing.) | | |
| Informational Presentations (10%) | 409 | 1639 |
| Training Workshops/Classes (15%) | 277 | 1756 |
| (Note: The 277 workshops last year reached over 4000 educators.) | | |
| Meetings For Local Staff (5%) | 65 | 623 |
| Newsletter Issues Published (5%) | 31 | 175 |
| Courseware Development (15%) | | |
| Daily Consulting To Individuals (15%) | about 12,000 inquiries per year | |
| Other (planning, conferences, contests, staff meetings) (15%) | | |

Even though MECC Instructional Coordinators are assigned to work with certain user constituencies, some of the work they do results in benefits for all users (e.g. courseware development). Thus, the assistance received by a particular user comes sometimes from "his own" MECC Coordinator and sometimes from the MECC team effort. The figures below show roughly the frequency with which a user receives the service in a particular area from the MECC Coordinator particularly assigned to his institution:

| | |
|---|---|
| Response To Daily Problems and Requests | (95%) |
| Requested Training Sessions (from a particular user) | (80%) |
| Open Training Sessions (any user may attend) | (20%) |
| Courseware Development | (10%) |

The fact that different members of a staff have different backgrounds and talents means that no individual will be able to meet all the needs of his constituency. The MECC staff members pool their talents so that almost any user need can be met by someone on the staff, time allowing.

## The Case For Centralization

The previous section touched on the idea that the sharing of expertise found on a staff of 10-12 people provides some common benefits for all users served. It is not mandatory that this type of service be provided by a centralized staff. It would not be unusual for individual school districts in a given geographic area to each have their own user services resource instead of a centralized team. A state that is divided into several intermediate service areas for provision of certain common services to districts in each area may choose to assign a classroom computing resource person to each area, reporting to the individual area administration.

However, the Minnesota experience has demonstrated that the centralized model on a statewide basis has some important benefits, as listed below:

1) there is a focal point for sharing user suggestions and contributed computing software across the state;

2) an organization with national visibility can keep tabs on the product market and efficiently distribute information on new high quality offerings to users;

3) changes in technology can be explored, evaluated, and implemented on a statewide basis in a simultaneous, coordinated manner;

4) individual educators whose local computing support isn't sufficient to cover their needs for either training time or expertise have a resource at a higher level to turn to for help;

5) local district and campus computing coordinators have a resource with special expertise from which to obtain the further advanced training they need to help their local staffs;

6) the problems involved in developing curriculum and training materials for computers can be solved in team fashion so that advanced techniques can be developed once and then used and disseminated widely.

Staff members that spend 8 hours (and usually more) a day responding to a never ending flow of user requests need colleagues within their own organization to fall back on for help. They need internal staff support to help solve problems outside their own area of expertise, to assist them with larger projects such as conference planning and courseware development, to share the immense task of keeping up with the ever increasing quantity of literature in the instructional computing field, to lift certain bureaucratic burdens from their backs, and to help keep their collective morale at a high level. These needs cannot be met effectively when the computing resource person is made part of an organization in which he is the only staff member working in the computing field. The centralized approach of creating a team of user services specialists to serve a large clientele, such as a whole state, provides benefits in a way that makes the whole greater than the sum of the parts.

## A Training Network

The creation of a statewide centralized user services organization does not mean that particular staff will provide all training to educators in the use of computers. Nor does it mean that educators will receive all of their help at the statewide level from the user services organization. A statewide resource such as MECC's User Services Department is only one component, though an important one, of an entire support network, as described below:

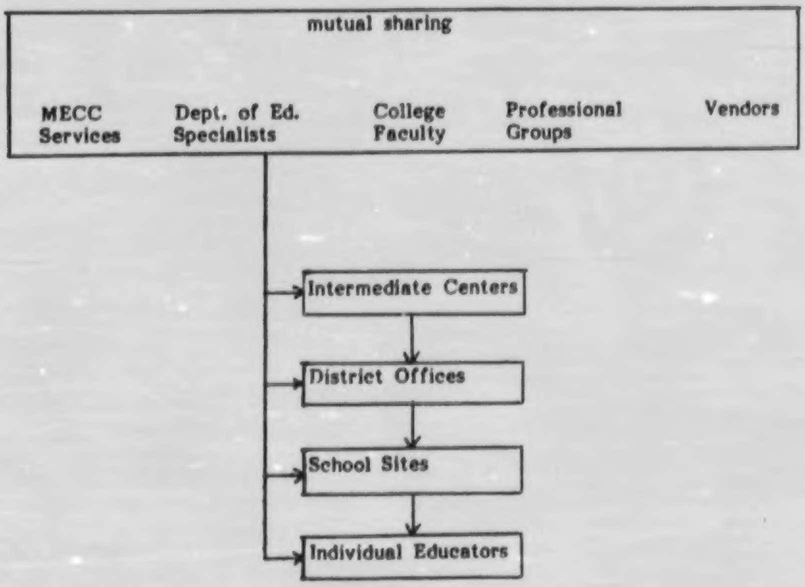
Sources of Statewide Instructional Computing Support

1) MECC User Services (and other departments within the MECC Instructional Services Division: Instructional Systems Design, Technical Services)
2) subject area specialists within the State Department of Education
3) computing and education faculty at colleges and universities in the state
4) professional organizations for educational computing and various subject areas
5) vendors of hardware and software.

Recipients of Statewide Instructional Computing Support

1) intermediate service centers - in Minnesota, the state's nine Educational Cooperative Service Units are getting involved in software review and bulk hardware purchasing for the school districts in each of their territories;
2) district offices - larger school districts which have full-time subject area specialists for traditional curriculum areas have in some cases added full-time staff to support classroom computing;
3) school sites - larger buildings have sometimes designated staff to coordinate computing activities at that level;
4) individual educators.

This network could be diagrammed as shown below. The diagram indicates that the network is not so rigid that statewide resources cannot be applied directly to any of the lower levels.



## The Evolution of User Services

The nature of a user services staff in support of instructional computing, like that of any organization, will change over time. As the users served become more knowledgeable, the services staff will have to steadily increase its expertise in many areas. Whereas the initial years might be spent training people how to operate equipment and software, users will eventually begin to desire information on how to integrate the computer into the total curriculum. As more educators become involved in computing, it will become less feasible for the central staff to work with individuals at the classroom level. The central staff might begin to turn more towards a role of "trainer of trainers", working with people at the district level and above who can in turn work at the classroom level. Along this line, the central staff might allocate some time away from face to face training to publishing training materials that others may use to do the job.

Initially, the people to be served will welcome any activity which teaches them more about computing. The services staff will probably put on many single session presentations, perhaps an hour in length, just to introduce computing to those with no experience. However, educators will eventually come to demand more formalized training in the form of classes that might run for five to ten two-hour sessions in covering a topic in depth. The emphasis will shift from simply gaining knowledge to acquiring actual skills.

Finally, a user services staff, which will probably start out with a high degree of spirit and comraderie to match the high demands of the work, will have to face the fact that good people don't often stay settled. Because a team operation provides an excellent training ground, talented services staff will within three to five years begin to receive offers to take their skills and abilities to other settings, especially in an area as fast growing as computing. Maintaining a level of excellence in the face of staff turnover is perhaps the most critical challenge to be faced. The ability of an organization to draw good staff replacements from the people who have been receiving its services is the most revealing testimony to the quality of its work.

## Summary

Instructional computing requires a large amount of support if educators are going to make significant use of it. This type of support is the job of a user services staff. User services covers a wide variety of topics, activities, and staff attributes. On a large scale, the effectiveness of user services is enhanced through a centralized team approach which is an important component in a total training network. The role of user services will tend to evolve along with the trends in technology and the needs of those served.

# COMPUTER LITERACY
# IN THE ELEMENTARY CLASSROOM

Jean Rice
Computer Consultant
Minneapolis, MN

Sandy O'Connor
Nat'l Cash Register
San Jose, CA

ABSTRACT: Contemporary trends in our society clearly emphasize the need for increasing the understanding of computers by people of all ages. This paper discusses a means for introducing computer literacy to young children based on seven years of experiences using the computer in the primary and upper elementary school classroom. Few learning tools have been introduced into the classroom which have stimulated the intensity of motivation for learning for all students as the computer. Today's children will be tomorrow's scientists, leaders and teachers. They are now growing up in a world of computers. They have no fears and apprehensions but rather are excited and fascinated by this electronic wonder. Their spontaneous enthusiasm is a delight for every teacher.

## WHAT IS COMPUTER LITERACY?

The definition of computer literacy today is changing in a dynamic way. The Nattional Council of Supervisors of Mathematics has listed computer literacy as one of the Ten Basic Skill Areas and in 1977 stated "students should know about the many uses of computers in society and they should be aware of what computers can do and what they cannot do."[8]

David Moursund in his point paper[7] for the National Computer Literacy Conference (December, 1980) discussed how "computer literacy initially tended to mean a level of understanding at which the student could talk about computers but could not actually work with the computer (now called computer awareness)." He further suggested that now computer literacy is "a working knowledge of computers. This knowledge should be at a level compatible with the other knowledge and skills that a student is acquiring in schools. It is a knowledge based upon understanding how computers can help one learn, how computers can help one solve problems, how computer knowledge is essential to a modern understanding of some topics, what is included in the field of computer and information science, and computers as a form of entertainment."

Kenneth Brumbaugh in his point paper[2] for the National Computer Literacy Conference noted "most individuals, including those who are in the computer field, are not comfortable and confident of their definition of computer literacy." He suggested "a definition of this concept be as follows: to be computer literate one must be able to define, demonstrate, and/or discuss how computers are used; how computers do their work; how computers are programmed; how to use a computer and how computers affect our society."

Beverly Hunter in her presentation for the National Education Computer Conference[4] (June, 1980) defined computer literacy functionally as "whatever a person needs to know and do with computers in order to function competently in our information-based society. This definition highlights the fact that specific skills, knowledge and values required will vary from person to person, from job to job, and from time to time. Specialized knowledge required for a career as a computer designer, programmer, technician or analyst is usually excluded from the computer literacy domain."

In an article from Educational Technology, December, 1980,[6] Colin MacKinnon described computer literacy as a "process" and further stated "the understanding of the language of the computing process and information handling is a dramatic challenge to the intellect and the ability of the mind to recognize, compare, analyze and synthesize fact and abstract concept as well."

## COMPUTER LITERACY PROJECTS

As a result of the increasing need for a computer literate society, various projects have been taking place around the country and materials are beginning to be developed to help fill the void in the school curriculum.

A set of computer literacy objectives (grades 7-12)[1] was the result of a study in Minnesota schools by the Minnesota Educational Computing Consortium (MECC) under a grant from the National Science Foundation in 1979.

Over 400, K-12 Course Goals in Computer Education were the product of a cooperative project of school districts in several Oregon counties by the Tri-County Goal Development Project[12] in 1979.

In 1980, the National Science Foundation awarded a grant to the Human Resources Research Organization (HumRRO) and the Montgomery County Public Schools to develop a curriculum guide for computer literacy, K-8.[4]

In addition, publishers and software vendors are now becoming involved in the development of computer literacy materials.

## COMPUTER LITERACY--INFUSION VS. UNIT INTEGRATION

How and where does computer literacy fit into the school curriculum? There still does not exist among educators in the field a concensus on this issue.

The approach being taken by HumRRO in the development of their curriculum guide is "to establish a process for infusing computer literacy objectives into the existing curriculum, using whatever materials and equipment are available."

The other approach the authors of this paper will discuss is a computer literacy unit being integrated into the curriculum.

HumRRO has described the advantages and disadvantages of their infusion process. They express that "the most compelling argument for the infusion approach is that it attempts to provide tools to use in all subject area learning in a manner and context that is clearly relevant to the students work."

They point out "the limitations and problems with the infusion are first of all that a school district must have a strong commitment from the Board of Education and the administration in order to provide the policies and resources needed to establish goals and implement systematic curricular change. In addition, it requires participation of many people in a school district and a basic level of computer literacy on the part of curriculum specialists and teachers. Even in schools involved with computers, many specialists have not yet achieved that level of literacy themselves."

The other approach, which the authors of this paper will describe to you, is integrating a computer literacy unit into the existing curriculum. One advantage of this approach is that the student has the opportunity to become familiar with the computer in one setting rather than fragmented over many areas. The student then is provided with a sound background of computer concepts before he or she moves into other areas using the computer. Many question whether simply using the computer in the classroom with limited exposure to computer terminology as well as other facets of computer literacy adequately provides a basis for becoming computer literate.

The teacher, new to computer concepts, also has the opportunity of becoming computer literate in one setting rather than being asked to use the computer in various subjects without a background of the capabilities, limitations and overall functions of the computer. One might argue that, at the present state of the art, the unit integration approach probably is the most expedient way to go now until computer literacy prevails among teachers.

A disadvantage is that the computer literacy unit approach might become a crutch which the infusion approach of the future may then find difficult to set aside.

Both of these approaches work. Whichever is most expedient to a particular situation is the one that should be used because of the importance of getting started.

## COMPUTER LITERACY IN THE ELEMENTARY SCHOOL

A few years ago the computer was found only at the college level. Today its presence is growing at the elementary and secondary levels all over the world. The elementary level is the ideal place to introduce the computer. A child's open, uncluttered mind is most receptive to using the computer and understanding its operation.

### Primary Grades--Computer Literacy[11]

The primary computer literacy unit introduces kindergarten through grade three students to beginning computer concepts. The unit may be used with or without computer equipment.

The teacher's guide is divided into the following sections:

A. Suggestions for computer users
B. Unit outlines for K-3 grades
C. Individual chapter aids
D. Project and activity masters
E. Films and suppliers
F. Resource books
G. Glossary

### Suggestions for Computer Users

This section includes suggestions for becoming comfortable with the computer, for selecting computer programs and for including parents and older students as assistants.

### Unit Outlines

Unit outlines include a minimum time frame for each grade level. Twenty minute class sessions of instruction and activities are suggested for each grade level. The number of class sessions varies from ten for kindergarten to twenty for grade three. In addition, two computer sessions for kindergarten and grade 1 are suggested and five sessions for grades two and three. (Computer sessions are from five to ten minutes each.)

### Individual Chapter Aids

The chapter titles within the unit are the following:

Computers Help Us
Computers Long Ago and Today
A Computer Has Many Parts
Microcomputers Are Small Computers
Flowcharts Are Computer Maps
Programs Tell the Computer What To Do

The material covers places where computers are used (in microwave ovens, traffic lights, stores, robots, etc.), ways of counting (stones, abacus, calculator, computer) and computer parts (keyboard, printer, memory, etc.)

A student will be able to draw a flowchart and to identify a computer language upon completion of the unit.

Objectives are included for each chapter with numerous suggestions for class discussions, projects, and activities. A chapter summary and crossword puzzle quiz with answer key are also part of each chapter.

### Project and Activity Masters

These masters accompany each class session. The students may work together on a project such as building computers out of cardboard boxes using paint, fabric, and buttons; or, they may work individually on an activity such as drawing a product bar code.

### Resources

Films and suppliers are listed to help the creative teacher expand the unit. A list of resource books and publishers is also included.

### Glossary

The glossary defines computer terms in simplified language.

The unit is especially designed for those with little or no computer experience and allows expansion for those with computer background. Its uncomplicated vocabulary is suited for K-3 use. The teacher suggestions and class activities are intended to make any teacher comfortable and successful teaching the unit.

## Upper Elementary--Computer Literacy[9]

Once computer concepts are introduced at the primary level, these concepts are then reinforced and expanded upon at the upper elementary level.

The upper elementary unit is designed to be used in a minimum time frame of fifteen 45-minute sessions. To cover all the projects and activities would require at least twenty eight sessions.

The unit can be used with or without a computer (note designated unit chapters* below). However, as discussed earlier, to be considered computer literate one should have a working knowledge of the computer, achieved only by using one.

The unit includes discussion of both large-scale computer systems with time-sharing capabilities as well as the smaller microcomputers.

### Student Text

Chapter titles within the unit are the following:

*What Is a Computer?
*How Are Computers Used?
*History of the Computer.
How Computers Work.
Devices Used To Get Information Into and Out Of a Computer.
*How To Plan a Computer Map (Flowchart)
*What Is a Microcomputer and How Does It Work?

### Teachers Guide

The teacher's guide which accompanies the student text contains supplemental background information and suggestions for class discussions. Pre and post tests are included as well as a student interest questionnaire for an assessment of understanding.

Duplicating masters of projects and activities are also contained within the guide. Students keep scrapbooks of such projects and activites as the following: a report of an interview with someone who works with computers, pictures collected from computer stores and companies (addresses listed in the guide); computer book reports (resource books listed in the guide); report of microcomputer comparisons; list of a Computer "Bill of Rights"; flowcharts; simple programs written by the student and a glossary of terms.

Suggestions are given for scheduling individual student time at the computer. Each school situation will vary with the availability of computer equipment. In situations where only one computer is available for a school, students may have access to the computer once or twice a week. A teacher may choose to have students input their own programs, take quizzes on the chapters in the book, or use other programs from the school's library of computer programs.

As part of the unit activities, guest speakers are invited into the classroom. These sharing experiences have proven enriching for both the students and the guest speakers. The speakers have been surprised at the level of understanding of computers by the elementary students. They have oftentimes found it easier to talk to them than to many adult groups because of the students familiarity with the terminology.[10]

Tours of computer facilities are suggested as part of the unit.

## Programming

Although only an introduction to programming is part of the computer literacy unit, many students have indicated a desire to learn more programming. In many schools, computer programming is offered to elementary students. Since a mathematics background is not a prerequisite for computer programming (only a logical and thinking mind), these young students have proven time and time again that they can write programs, in many cases equal to or better than older students and adults.

With the introduction of the microcomputer into the home and the school classrooms, opportunities for learning programming and other computer skills at the elementary level as well as at the secondary level will, of necessity, need to be provided. These may take the form of individualized self-learning opportunities or scheduled group classes. As has already been proven, children at some point tire of the game playing activities on the computer. They soon move into learning how to write their own games. They desire the challenge of higher cognitive activities.

Laying a good computer foundation in the elementary grades readies students for the opportunity to move into more sophisticated computer activities at the secondary level.

## IN CONCLUSION

Why the need to teach computer literacy? Professor Joseph Weizenbaum of MIT says, "There is the possibility of people absolving themselves of responsibility for anything which the machines emit. The question is if the computer gives you advice, do you just follow it blindly? When people don't really understand what's going on behind the scenes--such as inside a telephone exchange--it becomes rather like magic. So who's going to evaluate the device? The possible result--don't blame me. Blame the machine."

Colin MacKinnon notes by adding computer literacy to the list of other skills acquired in the educational environment, "we may assure the continuing growth of the intellect and the advancement of culture. Possibly, too, we may sweep aside the traditional misconception that the computer will eventually do our thinking for us."[6]

The computer is a tool to help people function better in today's world. Until our society is computer literate, however, the full potential of the people using the computer will not be optimized.

Bringing the computer into the school curriculum provides a wealth of new opportunities for learning. The computer will never replace the teacher. As Andrew Molnar states "Any teacher who thinks he or she might be replaced by a computer, should be."[5]

Ours is a world ready for computer support. We are seeing only the tip of the iceberg. Thomas Lee at Bell Laboratories has said, "The most exciting applications will not come until the kids who are still in high school and have grown up with pocket calculators and home computers become the engineers of the 1980's and 1990's."[3] The Economist noted "it is estimated at least 25,000 applications of the computer await discovery. To ask what the applications are is like asking what are the applications of electricity."[3]

This is 1981 and it is happening!

# REFERENCES

1. Anderson, Ronald & Klasson, Daniel L, Computer Literacy--What Is It? Mathematics Teacher, February, 1980, Vol. 73, No. 2, pp. 91-96.

2. Brumbaugh, Kenneth E., Computer Literacy--1985, Point Paper for the National Computer Literacy Conference, December. 1980.

3. ....The Computer Society, Time, February 20, 1978, pp. 49, 53.

4. Hunter, Beverly, Human Resources Research Organization (HumRRO), An Approach to Integrating Computer Literacy Into the K-8 Curriculum, Paper for the National Education Computer Conference, June, 1980.

5. Koetke, Walter, Educators Are Buying, Microcomputing, January, 1981, pp. 14, 15.

6. MacKinnon, Colin, Computer Literacy and the Future: Is it Possible to Prevent the Computer From Doing Our Thinking For Us? Educational Technology, December, 1980, pp. 33-34.

7. Moursund, David, Personal Computing for Elementary and Secondary School Students, Point Paper for National Computer Literacy Conference, December, 1980.

8. NCSM, National Council of Supervisors of Mathematics Position Paper on Basic Skills, January, 1977.

9. Rice, Jean, My Friend the Computer, Student Text and accompanying teacher's guide, T. S. Denison & Co. Minneapolis, 1981 (revision)

10. Rice, Jean & Haley, Marien, My Computer Picture Dictionary, T. S. Denison & Co., Minneapolis, 1981.

11. Rice, Jean & O'Connor, Sandy, Computers Are Fun, Student Text and accompanying teacher's guide, T. S. Denison & Co., Minneapolis, 1981.

12. Ricketts, Dick, Project Director, Oregon Tri-County Goal Development Project; K-12 Course Goals in Computer Education, 1979.

# IMPLEMENTATION OF A
# COLLEGE-WIDE COMPUTER LITERACY PROGRAM:
# A WORKING MODEL

Barbara Rich, Assistant Dean
Marymount Manhattan College
221 East 71 Street
New York, N.Y. 10021

Marymount Manhattan College is a private, urban, Liberal Arts College for women. The mission of the College is to educate the socially and economically diverse population of the New York City area by fostering intellectual achievement and personal growth, and by providing opportunities for career development.

One of the College's objectives is to achieve a level of computer literacy among students, faculty and staff so that they may function more effectively in a society which has become increasingly dependent upon the computer as a means for generating and transmitting knowledge.

A program has been designed and is being implemented at Marymount Manhattan College to help develop computer literacy among all members of the College's community.

## Introduction

In 1980, the concept that computer literacy should be incorporated into a Liberal Arts education is still innovative. The need to support this integration becomes clearer every day as society relys more and more on the computer for transmitting and generating knowledge.

Marymount Manhattan College is a private, independent, urban, Liberal Arts college for women in New York City. It has over 2,000 full and part time students and 52 full time faculty. The College has committed itself to implementing a college-wide computer literacy program. It has defined computer literacy as the understanding by the learner of the uses of the computer, the capabilities and limitations of the computer, and the ability of the learner to utilize computer instruction and data analysis packages.

The College's decision to inaugurate a computer literacy program was an outgrowth of its mission, its goals, its student body, and its faculty.

The mission of the College is to educate the socially and diverse population of the New York area by fostering intellectual achievement and personal growth and by providing opportunities for career development.

The majority of traditional-aged students come from urban high schools which lack the resources necessary for computer support in education. For the returning adult students, computers in secondary education came at least a decade or more past their high school graduation.

Many of the students have continued to avoid mathematics and computers because of their anxiety about competency in these areas.

At the inception of the program the majority of faculty and staff were, with few exceptions, also unfamiliar with the uses of a computer.

During the past two years the College took five steps which helped put the program in place:

1) After two years of discussion, the faculty adopted a statement reflecting its commitment to the Liberal Arts as the vital center of all the College's programs in the arts and sciences.

2) In January, 1980, at the Long Range Planning meeting of faculty and staff, the College reaffirmed its mission and approved the implementation of a Computer Literacy Program as a College objective.

3) The College became a participating institution in the Council of Small College's, Quality of Education Project (CASC Que). It selected the implementation of computer literacy for students and faculty as its project. The project involves the direct participation of the President of the College and a faculty-

administrator-student campus team. It has helped define the time-line of the program, and its needs and outcomes.

4) A computer literacy committee has been appointed. It includes members of the CASC Que Project Team and additional faculty members. It is responsible for formulating new plans and for reviewing components of the existing program in order to make recommendations to the Faculty Council and the Vice President of Academic Affairs.

5) The College purchased a mini computer in the Fall of 1980 for computerizing the Administrative Management Systems which will involve detailed training of the administrative staff.

The Computer Literacy Program as it is now developing reflects planning and continuous evaluation. While it goals are clear, the restraints of budget, time and personnel are frustrating. But these constraints have alco stimulated new ideas, as I think the program will reflect.

The Computer Literacy Program

The program reflects the needs and desired outcomes for students, faculty and staff. As outlined in the interim CASC Que report[1] the needs are:

1. Current State: about 98% of Marymount Manhattan College's students are computer illiterate.

Desired State: To increase levels of computer literacy by 20% each academic year.

2. Current State: There exists a substantial computer anxiety, which may be an off-shoot of math anxiety, particularly evident among women students.

Desired State: Reduction of this anxiety.

3. Current State: The majority of faculty are not computer literate.

Desired State: To raise the level of computer literacy among faculty so that they can employ computer instruction in courses throughout the six dividions of the College.

Anticipated student outcomes are:

1. Students will be able to use the computer to perform a task in the student's area of study.

2. There will be greater comfort in computer use as measured by the increased number of users and increased choice by students of elective courses where there is a computer component.

3. Some students will consider further study in both computer, mathematics and science areas.

Anticipated faculty outcomes are:

1. More faculty will incorporate computer assisted techniques in their courses.

2. Computer literacy will enable faculty and students to handle real data rather than text book problems in such courses as: Calculus, Sociology, Psychology, Statistics.

3. The program will provide faculty with a tool to help underprepared students by providing them with pre-programmed drill material.

4. The program will assist faculty in their new areas of research.

To meet these needs and obtain these outcomes, the College has designed a program which involves training, personnel, course development and equipment.

Faculty support and training are critical to the success of any computer literacy program. The hypothesis being that faculty who understand the potential uses of the computer as a teaching tool, who feel comfortable using the computer, and who understand the teaching programs available to them will implement computer use in their courses. They will encourage students, through the advising process, to enroll for these courses and for courses in the Data Processing Sequence.

Two years ago faculty training seminars were instituted. These seminars in BASIC are given in one week sessions. A specially designed "computer use' course was also offered this past Fall in an eight-week period. Faculty have indicated that one week courses better fit their time demands. Staff have also attended these courses.

Students are being encouraged to enroll for the BASIC course or the Introduction to Data Processing course. The Introduction to Data Processing, which was first offered in the Fall of 1980, is heavily enrolled. Many members of the staff will be trained via coordinating groups which have been organized to train them for the installation of computerized management systems. Staff includes admissions, advisement, registrar, and student accounts personnel.

Marymount Manhattan College has a 21-credit Data Processing Certificate Program, registered by the State Department of Education of New York. Basically, it calls for 15 credits of requirements and six credits of elective. The requirements include courses in FORTRAN, COBOL, Assembler Language, and System Design and Analysis.

The program was designed for matriculated students at the College and for professionals in the community who need or want the certificate as part of professional development.

Courses in Chemistry, Psychology, Developmental Mathematics, pre-Calculus, Calculus, Algebra and Sociology are already using the computer in a CAI, tutorial or demonstration capacity. Plans are being formulated to expand computer use within the next two years in Physics, Communications Arts, Philosophy, and Education. It is expected that as more faculty continue to take the training, these course projections will increase.

The Accounting and Business Management majors now include a Data Processing requirement for their students.

Students have been hired as laboratory assistants for the newly allocated Computer Lab. They will assist in helping fellow students in operating the computers.

In order to develop this program equipment and space become critical issues.

The College originally had a time sharing arrangement with another institution. As the costs rose, and the program grew, this arrangement no longer seemed practical. Through funding provided by a Title III grant, the College purchased 10 microcomputers. An additional microcomputer was purchased through funds from a faculty development grant.

Currently the microcomputers are housed in a computer lab in which the Data Processing courses are held, the mathematics lab, the library's audio-visual room, the natural science and mathematics and sociology divisions.

It is too early to evaluate the program's success. We have learned that there are problems yet to be resolved:

1. How to provide faculty adequate time to train, plan and implement computer use in their courses given their time realities for teaching and committee work.

2. How else to address the reduction of computer anxiety among students.

3. How to provide space and equipment as the program expands and the budget doesn't.

These questions have been raised and will be addressed through the continued planning and evaluation process of the Computer Literacy Committee, the faculty and administration.

What is clear is that the need to continue the program is not in question. It is a program designed to meet the educational and professional needs of the students and as such is an integral part of a Liberal Arts education.

References cited:

[1] Rich, Barbara, Interim Report to CASC QUE, February, 1981, Marymount Manhattan College, N.Y.

# FOR COMPUTER-BASED EDUCATION TO MATTER IT NEEDS CRITICAL MASS

Peter J. Rizza, Jr.
Senior Educational Consultant
Education Technology Center
Control Data Corporation
(P.O. Box 0, Minneapolis, MN 55440)

It is generally accepted that education will change due to the influx of new technology into the teacher-learner process. While new technologies have always been investigated by educators, they have rarely been embraced. The concept of Computer-Based Education (CBE) has matured over the past decade to a realistic expectation, but is not widely implemented. Why is this the case?

The implementation of a high-quality computer-based education system is a complex task that can only be successfully completed when the right mix of all of the proper ingredients exists. This paper investigates the four major elements required for educational and computer technology to merge into a viable CBE system. The paper describes the four major ingredients required to make up the mass of a computer-based education system. The author explains that the presence of these four ingredients alone does not constitute a CBE system which can be implemented successfully. For the CBE mass to become critical, the ingredients must exist in necessary quantities and qualities. When this happens, an explosion can take place and CBE can become more than a new concept, it can become a reality.

## INTRODUCTION

It is estimated that the total amount of knowledge existing in a given academic area increases at a rate of 100% every 25 years. This information explosion has obvious implications for the way students become educated. Students can no longer be expected to spend their school years simply absorbing information, they must learn to organize knowledge in such a way as to survey the information base and select those facts and principles which are most meaningful. Over the years, teachers have been the key link in the chain of education bringing knowledge and students together. Today, the teacher is faced with new problems because the amount of knowledge is increasing rapidly. In the development of curriculum, how does an instructor decide which piece of information to include and which to exclude? How can more of this information be transmitted to the students in the same amount of instructional time? These questions and many more have evolved into major considerations for instructional development in the past few years. In response to the need to transmit more meaningful information to learners in the least amount of instructional time, educators and curriculum developers have developed and introduced many new techniques to raise the efficiency and the effectiveness of the instructions delivered. Many of these new techniques have been the result of a merger of education technology and computer technology called computer-based education (CBE). (See Figure 1.)

## EDUCATION TECHNOLOGY

There have been many advances in the field of education which fall under the umbrella of educational technology. The concept of individualized instruction was introduced to allow more students to progress through different types of instructional materials. Individualized instruction is an attempt to be efficient with the learner's effort and time. Individualized instruction introduced diagnostic testing for better student placement, behavioral objectives with criterion-referenced tests for more accurate student assessment, and mastery learning models for higher overall student performance. The tools of formative and summative evaluation were developed to better assess the effectiveness of the instruction and provide data for improvement of the learning system. Finally, the concept of an instructional systems design was developed to focus on new organizations of the instructional material and better logical structuring of the curriculum. Other improvements were made in the areas of instructional delivery. Most notable was the use of computers to aid in the educational process.

## COMPUTER TECHNOLOGY

The growth of computer capabilities over the past few decades has influenced modern society in many ways. The fact that, in a matter of minutes, computers can place a large volume of data at one's fingertips for analysis and detailed investigation has changed the way many companies, institutions and organizations function. It is a general rule of thumb within the computer industry that computing power will double every four years while the cost of delivering that computer power will be cut in half during the same time period. This ever-increasing capability of the computer to support organizations such as educational institutions, coupled with the continuing decrease in actual costs to deliver this type of support, has moved educators and computer scientists into common problem-solving activities. The computer has become well established as an aid in education administration, and more recently has been introduced as an aid to instructional delivery and management as well. Specific applications of computers to the learning process have evolved in sophistication from simple, mechanical presentations of programmed instructional text to the delivery of very complex courseware structures. Many of these CBE courses allow for immediate student feedback and continuous management of learning prescriptions. The field of computer-based education (CBE) has grown, however, far beyond the simple merger of hardware (the physical computer equipment), software (the computer programs), and courseware (the subject matter material stored in the computer), into a total educational environment. This environment is characterized by uses of educational and computer technologies specifically designed to aid in the learning process.

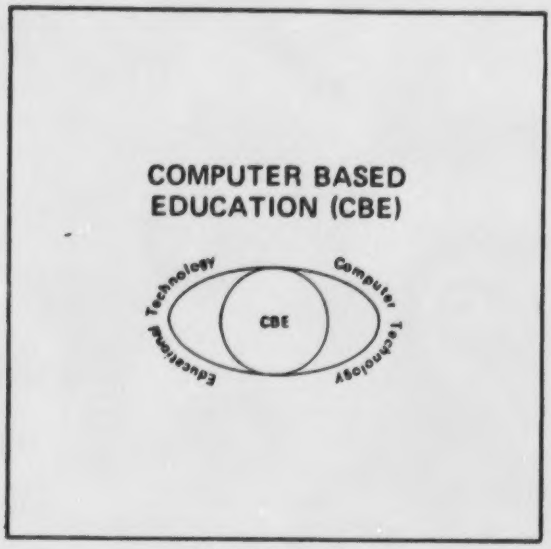


FIGURE 1.

NEEDS OF EDUCATION

Today, educators are faced, more than ever, with problems of providing quality education. The needs of education are in the areas of accountability of the learning process, higher quality of instructional delivery, lower cost of instructional delivery, efficient and effective ways of delivering instruction to a wide variety of student populations, and the ability to motivate student population to want to learn.

Most of the CBE systems in operation today are aimed at solving one or more of the needs just mentioned. Some CBE systems focus on academic control to keep track of the students as they pass through the learning instruction and present student record data to the educators to aid in evaluating performance. This is referred to as computers for academic control (CAC). Some CBE systems focus on instructional delivery which is, of course, the requirement of any learning system. This is referred to as computer assisted instruction (CAI). Other CBE systems focus on instructional management which is required of the learning system, particularly if they are designed to be flexible and efficient of the student's time and the educator's money. This is referred to as computer managed instruction (CMI). Still other CBE systems focus their applications as learning resources to aid in helping the learner in the actual learning process, not as a deliverer of instruction but as a computer resource to the delivery of instruction. This is referred to as computer supported learning aids (CSLA). These four major categories of CBE applications are illustrated in Figure 2. Through the selective application of the computer to meet the needs of education, a comprehensive CBE system could control, manage, instruct, and support the educational environment in its entirety.

It is generally accepted that education will change due to the influx of new technology into the teacher/learner process. While new technologies have always been investigated by educators, they have rarely been embraced. The concept of computer based education (CBE) has matured over the past decade to a realistic expectation but is not widely implemented. Why is this the case?

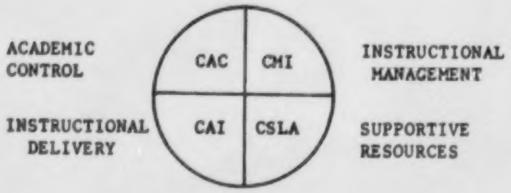


FIGURE 2.

ACCEPTANCE OF CBE

As noted earlier, computer-based education techniques have been developed to address most of the pressing educational needs. These techniques have, in many cases, proven to be educationally effective and economically sound. Why have not these fine examples of CBE systems been embraced by the educational community? Why hasn't CBE become a wide-spread, accepted delivery mechanism throughout the education industry? There appears to be certain barriers to the receiving of the CBE techniques. These barriers come from meeting what can be called the "critical mass". This is a metaphor drawn from the concept in physical science which describes what happens when the necessary ingredients exist in proper proximity in the proper quantities — a critical mass exists and a greater reaction takes place. The implementation of the high-quality, computer-based education system is a complex task that can only be successfully completed when the right mix of all of the proper ingredients exists. There are four main ingredients that are necessary before computer-based education techniques can be adopted.

Hardware: Computers must exist in a quantity that will be able to handle the system designed for computer-based education. In the United States, there is no lack of powerful, large-scale computers and small-scale computers. There is more computer-hardware power per person in the United States than anywhere else in the world. Yet, access to the computer hardware is not very common for most school systems. While there is sufficient hardware to support computer-based education throughout the United States, a reassessment of how available hardware could be brought to bear on the task of supporting computer-based education should be investigated. With the introduction of the micro-technology into the computer industry, the availability of very small, high-powered microcomputers to the consumer is becoming commonplace. However, these readily available, inexpensive microcomputers must be evaluated in terms of their capability to support the solution to the educator's problem.

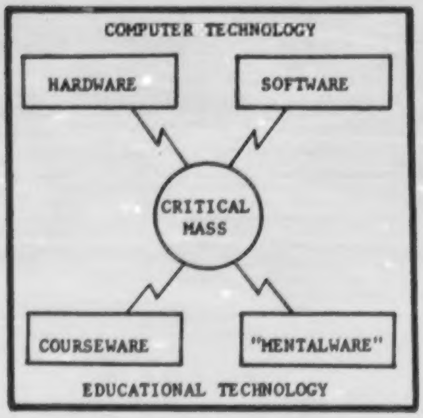
Software: The software for the computers has been, in the past, oriented toward commercial/business applications. While other software capabilities have evolved in the area of science and research, very little software has actually been developed for educational purposes. It is much easier to locate administrative applications on computers for schools than instructional applications, because the administrative applications parallel the already-existing business and commercial software which exists in great quantity and a variety of forms. With the introduction of systems like the Control Data PLATO system and other large-scale systems specifically designed to support educational purposes, the void in educationally oriented software is being filled. Many of the new microcomputers are also capable of delivering educationally oriented applications due to the very nature of the software developed for the micro, which tends to be individual/consumer oriented. Even with these developments, most software used in education is merely a slight modification of a business program.

Courseware: It is this ingredient that links instructional systems design with computer-based education. The courseware component represents the marriage of education and the computer technologies. To date, very limited courseware packages are available in the marketplace. Control Data has invested heavily in the development of a courseware library for delivery on the Control Data PLATO system and represents the major effort in courseware development in the computer-based education industry. Many other vendors, including microcomputer companies, are expecting that courseware will be developed by their users and submitted for publication and dissemination. This tends to be an unsophisticated and unorganized approach to courseware development and the likelihood that major curriculum efforts will evolve in this way is extremely low. A comprehensive systems approach to courseware development, similar to the one used at Control Data, is advisable if large-scale curriculum efforts are expected to evolve.

"Mental-ware": This ingredient represents the concept of the mental characteristic in the implementers of computer-based education programs. There must be a predisposition to organize education in a systematic way. The administration of a school implementing a CBE system must be willing to provide the support resources necessary. School faculty must be willing to alter the "traditional" school day schedule to better take advantage of CBE offerings. Instructors must become "literate" not only in how a computer works but, more importantly, how a computer can aid in the education process (CBE awareness). Parents should help support schools which implement comprehensive CBE systems. Colleges of education must prepare new teachers to use technology in education and help inservice teachers to quickly assimilate new technology into their classrooms. In essence, the "mental-ware" factor is determined by how prepared the administration, teachers, parents, and students are to use new technology effectively.

## CONCLUSION

These four major elements are required before educational and computer technologies can merge into a viable CBE system. (See Figure 3.) The presence of these four ingredients alone does not constitute a CBE system which can be implemented successfully. For a school system to expect to implement a CBE system based primarily on the hardware and software capabilities of the new microcomputers, for example, will prove to be a mistake. The CBE system introduced in this manner will only survive when the courseware and mental-ware ingredients have been addressed and satisfactorily developed. For an educational institution to expect to implement a CBE system based on a tremendous amount of courseware available in a library for student access, without the proper mental-ware support, will prove to be less than adequate. Additionally, for a school system to train its staff in the mental-ware processes required to support a CBE system, without providing it with adequate hardware, software and courseware to access, will cause frustration and failure. For the CBE mass to become critical, the ingredients must exist in the necessary quantities and qualities. When this happens, an explosion can take place and CBE can become more than a new concept, it can become a reality.


COMPUTER TECHNOLOGY
HARDWARE
SOFTWARE
CRITICAL MASS
COURSEWARE
"MENTALWARE"
EDUCATIONAL TECHNOLOGY

COMPUTER-BASED EDUCATION

CBE INGREDIENTS NECESSARY
FOR CRITICAL MASS

FIGURE 3.

## REFERENCES

R. T. Heimer and P. J. Rizza, Jr., "Some Basic Principles for the Design/Development of Courses to be Delivered by Computers," Paper presented at the World Conference on Computers in Education, Lauzanne, Switzerland, July 1981.

J. E. Searles and P. J. Rizza, Jr., "Computer Based Education for Developing Educational Systems," Paper presented at ADCIS annual meeting, March 1978.

J. E. Searles and P. J. Rizza, Jr., "Cultural Diffusion of Educational Technology," EDUCATION TECHNOLOGY MAGAZINE, November 1976, pp 46-48.

# Instructional Design vs. Authoring of Courseware: Some Crucial Differences

M. D. Roblyer
ICON Enterprises
Tallahassee, Florida

The term "instructional design" was originally coined by learning theorists and military/industrial trainers to describe a model-based, systems approach to creating effective instruction for various technologies. In early development of courseware for the rapidly-expanding microcomputer market, many developers adopted the term "authoring" to describe a more generic development process. This paper will point out some crucial differences between authoring and instructional design approaches, and will describe a courseware design model and systematic procedures which can help assure that courseware packages facilitate efficient and effective student learning.

## Introduction

There is agreement among instructional computing experts that the future success or failure of microcomputers as a medium of instruction (as opposed to an object of instruction) hinges largely on the availability of quality courseware. And yet, one would have to be insensitive to the field not to observe that the majority of microcomputer courseware materials currently on the market is not of acceptable instructional quality.

What is the reason for this overall lack of quality? A major part of the problem can be attributed to the development methods used to produce much of the available materials. In most initial development efforts, courseware was produced by technical personnel, who viewed the production process as "programming," and the product as "applications software." While this is certainly one true perspective on courseware, personnel who held this view invariably failed to recognize the need for special expertise in designing the application part of the software package - the instruction itself. Consequently, much of the specialized knowledge we have gained over the years about creating effective instruction was never applied to microcomputer courseware development.

Even now, when it is widely recognized that programming expertise alone is not enough in courseware development, there still appears to be an emphasis on "authoring CAI programs," rather than on "designing instructional courseware." Although standardizing terminology is probably neither absolutely necessary nor feasible, it is evident that those who think of courseware in terms of authoring have far different procedures and products than those who view the process as instructional design.

An examination of the differences between authoring and instructional design methods will be done here in order to demonstrate how use of the latter can help improve the overall quality of microcomputer courseware. While use of instructional design techniques cannot guarantee that courseware will be effective, it will be shown how use of such methods greatly improves the likelihood that materials accomplish what the developers intended.

## Differences Between Authoring and Instructional Design of Courseware

Four major points of difference can be identified between many current development methods and those methods known as systematic instructional design.

Use of a model. While authoring procedures usually vary considerably from author to author, instructional design procedures follow a theory-based model. There are both hypothesized and proven benefits to employing such a model. First, as Gagne and Briggs (1974) point out, an important assumption underlying use of systematic design methods is that "designing instruction must be based upon knowledge of how human beings learn." The use of a planned, organized method of creating instruction which reflects appropriate "conditions of learning" for individual students has historical bases in the early uses of technology in education. (See Gagne and Briggs, 1974, page 210.) Models of instructional design contain step-by-step procedures for identifying exactly what individuals are to learn and specifying what activities are required to bring about that learning.

Use of an instructional design model also recognizes that creating individualized instruction to meet the needs of many individuals is a very complex process. A model acts as a management device to help the developer structure development activities and make sure that all important aspects of the process have been addressed.

Since instructional design methods have emerged from many different projects in various locations over the years, it is not surprising that there is more than one design model in use. All of the models, however, have elements in common. The purposes for employing them are the same (identifying learning outcomes and managing the process of development materials matched to those outcomes). In addition, Briggs and Wager (1981) have identified certain "anchor points" in the instructional design models. These include:

- Statements of behavioral objectives
- Specification of testing methods
- Specification of methods for developing materials.

One such model which has been developed especially for the process of designing courseware (Roblyer and Wager, 1980) is shown in Figure 1. Some of the specific procedures and products of this model will be discussed later in this paper.

Emphasis on written design document. In the past, it has often been characteristic of courseware authors to begin programming activities either immediately, as the first step in development, or after a brief written or oral description about what is needed in the courseware. By contrast, instructional design methods require several written products which specify in some detail the intended outcomes of the courseware and how the materials are to appear, before any actual development begins.

The need for written documents at this stage is three-fold. First, they help assure that the planned product is matched to an instructional need, and that the purpose of the courseware is clear to everyone, including the programmer. A step often neglected in all kinds of software development is a clear statement of the problem being addressed. Second, a design document helps check for a match between the problem and the planned solution. Finally, it provides a blueprint to the programmer, much in the same way as a builder uses an architect's blueprint, which may be followed in "building the courseware."

This document is done before and in addition to the programmer's flowcharts. Such flowcharts document the technical method by which the materials will be constructed, not the design of the materials themselves.

The products of the model shown in Figure 1 include:

- Statements of goals and objectives - These are always in terms of what the student is to do in the lesson, not in terms of what the computer or the courseware will do.
- Learning map - This especially necessary when the courseware package is a skills series or part of a skills series. It shows three things: all of the skills to be covered and those required as entry skills for each package; hierarchical or prerequisite relationships among the skills; and finally, a suggested sequence for presenting the various skills instruction.
- Description of learning events - Part of the problem with many existing courseware products arises when they assume that drill-and-practice is a kind of self-contained instruction. In fact, drill is merely an activity for one of the events of instruction. Gagne (1977) has delineated nine different events from which the designer may select and design activities matched to them. These events include:
  (1) Gaining attention
  (2) Informing learner of objective
  (3) Stimulating recall of prerequisites
  (4) Presenting stimulus materials
  (5) Providing learning guidance
  (6) Eliciting the performance
  (7) Providing feedback on correctness
  (8) Assessing performance
  (9) Enhancing retention and transfer

Some events in the same instructional package may be on-line in the courseware; others may be on other media. The important concern is that conscious decisions are made on which of these events to put on-line, and then an activity is designed

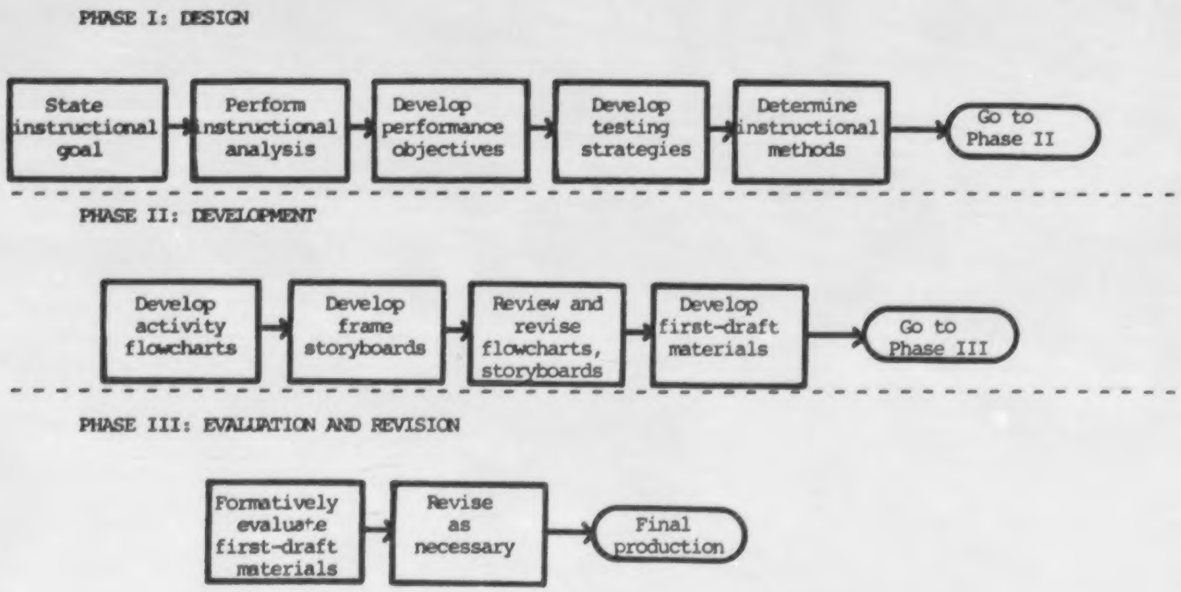

PHASE I: DESIGN
State instructional goal
Perform instructional analysis
Develop performance objectives
Develop testing strategies
Determine instructional methods
Go to Phase II
PHASE II: DEVELOPMENT
Develop activity flowcharts
Develop frame storyboards
Review and revise flowcharts, storyboards
Develop first-draft materials
Go to Phase III
PHASE III: EVALUATION AND REVISION
Formatively evaluate first-draft materials
Revise as necessary
Final production

Figure 1

Courseware Design Model

which will accomplish each event. The object then becomes to create useful instruction, rather than to develop a "tutorial" or "a drill package."

- Flowcharts and storyboards - These are documents in the development phase of the design process, just before first-draft programming is done. An example of an activity flowchart is shown in Figure 2. An example storyboard frame format is shown is Figure 3. A storyboard is produced for every frame to appear in the courseware.

Team approach to development. The term "authoring" carries a strong implication that courseware is done by one individual developer. There is considerable practical experience to suggest that a team of two or more specialists is needed to develop effective courseware (Gagne and Briggs, 1974). This team usually consists of an expert in the content (i. e. teacher of math, reading specialist, etc.), an instructional designer and a programmer. Sometimes, the content expert and designer are the same person. However, it is rare that an excellent programmer is also a trained instructional designer. Although it is not impossible that one person could do both design and programming, the serious design flaws apparent in much programmer-designed courseware produced to date is evidence that good programmers do not often make good designers. (And probably the opposite is also true.)

There are two primary reasons that at least a 2-3 person team be involved in the development of courseware. First, and most important, all of the expertise required for the development is there, and each part of quality control is being done by a specialist in that area. Secondly, there is insurance that the biases and errors of one individual are not reflected in the final product. Each member of the team helps by contributing to and reviewing what is produced at each step.

Provision for formative evaluation. A final implication of the term "authoring" which has probably resulted in more courseware deficiencies than any other aspect, is that "authoring" seems to describe only a development process. Instructional design methods include some provision for a formative review and revision phase (Dick and Carey, 1978). Perhaps more than any other step in instructional design, this kind of review can help assure that courseware accomplishes what the developers intended.

Several different kinds of review and evaluation may be done. First, subject matter and design experts from the design team or outside of it may preview the design documents (especially the activity flowcharts and storyboards) for content

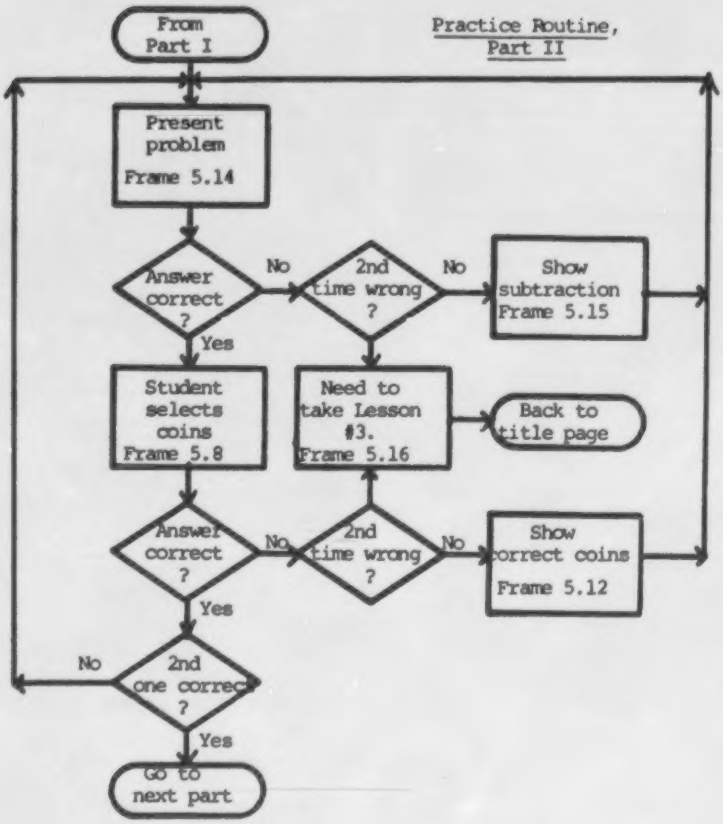
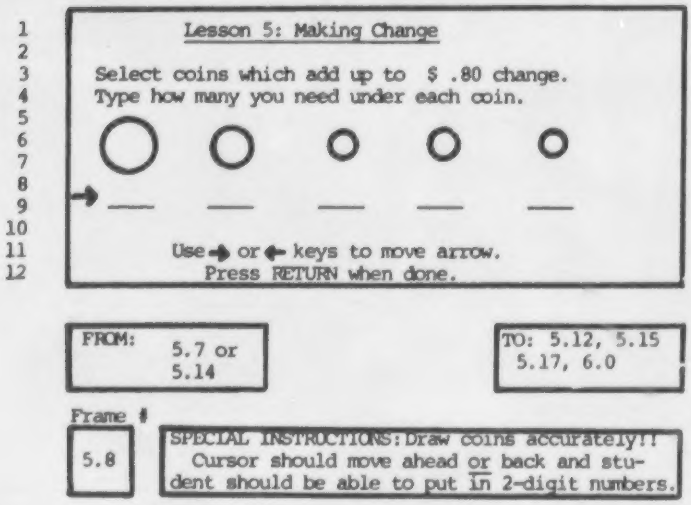

Practice Routine, Part II
From Part I
Present problem
Frame 5.14
Answer correct ?
No
Yes
2nd time wrong ?
No
Show subtraction Frame 5.15
Student selects coins Frame 5.8
Need to take Lesson #3. Frame 5.16
Back to title page
Answer correct ?
No
Yes
2nd time wrong ?
No
Show correct coins
Frame 5.12
2nd one correct ?
No
Yes
Go to next part

Figure 2

Example of Activity Flowchart



Lesson 5: Making Change

Select coins which add up to $ .80 change.
Type how many you need under each coin.

Use → or ← keys to move arrow.
Press RETURN when done.

FROM: 5.7 or 5.14

TO: 5.12, 5.15 5.17, 6.0

Frame #

5.8

SPECIAL INSTRUCTIONS: Draw coins accurately!! Cursor should move ahead or back and student should be able to put in 2-digit numbers.

Figure 3

Example of Storyboard Frame Format

accuracy and appropriate sequence, before any programming is done. Most kinds of formative evaluation involve actual use of the first-draft materials by students. Some useful formative activities are called:

- One-on-one use - In this evaluation, the designer sits with one or two students (usually one at a time) as they go through the instruction.

- Small-group use - This kind of activity involves a group of 10-20 students from the target population going through the courseware under more-or-less "real" conditions.

- Field-test use - Several groups from the target population go through the courseware in the field (classroom or laboratory where products are usually used).

The emphasis in all of these activities is on performance, rather than appearance of the product. Data are usually collected on the progress of the students in attaining the objectives of the courseware. Students and teachers are interviewed for their attitudes on using the courseware. The purposes of all kinds of formative evaluation are to identify any weaknesses which should be corrected before the courseware is released for general use. Naturally, a required follow-up activity to formative evaluation is a revision cycle in which the problems found during the review are corrected.

## Summary and Conclusion

Whether one uses the term "authoring" or "instructional design" to describe courseware development is not really important, although "authoring" is not a term usually associated with systematic methods of assuring instructional quality. This paper has proposed that, by whatever name it is called, four primary elements characterize an effective design process for creating quality courseware:

(1) Use of a theory-based model
(2) Emphasis on a written design document
(3) Team approach to development
(4) Provision for formative evaluation.

Perhaps the overall contribution of these four elements is to reverse some past trends in courseware development which have resulted in a large quantity of poor-quality, ineffective products. The first of these trends which must be reversed is the emphasis on courseware as program rather than as instruction which must be used by students. Instructional design methods can help switch this emphasis back to developing instruction.

Finally, the trend of developing courseware through isolated, unsystematic efforts of individuals must be changed. While it is possible to develop some courseware materials in this way, a large volume of microcomputer courseware must be produced in the next few years to meet the growing needs of both teachers and students. The sheer amount of development which must be accomplished in order to change the current courseware picture makes it essential that systematic instructional design methods be used to guide the process.

## References

Briggs, L. J. & Wager, W. Handbook of procedures for the design of instruction. Englewood Cliffs, New Jersey: Educational Technology Press, 1981.

Dick, W. & Carey, L. The systematic design of instruction. Glenview, Illinois: Scott, Foresman and Co., 1978.

Gagne, R. M. The conditions of learning. (3rd ed.) New York: Holt, Rinehart and Winston, 1977.

Gagne, R. M. & Briggs, L. J. Principles of instructional design. New York: Holt, Rinehart and Winston, 1974.

Roblyer, M. D. & Wager, W. The systematic design of computerized instructional materials. AEDS Workshop, Orlando, Florida, 1980.

# LONG-RANGE PLANNING FOR INFORMATION SERVICES AT LARGE UNIVERSITIES

Peter Roll
Office of Academic Affairs
University of Minnesota
Minneapolis, MN 55455

**Several indications are now beginning to appear that, at least for research universities and other large, complex organizations, the massive technological changes so long predicted by futurists have begun and must be incorporated into a 10- to 15-year planning framework for all aspects of information technology -- instructional and administrative data processing telephone and data communications system(s), electronic and other message delivery systems, library and data base services. Decisions that must be made soon in one or more of these areas will describe why a long-range planning framework is becoming so important to research universities and other large organizations (a primary reason is that a data terminal or microcomputer of some kind now seems likely to be found on the desk of most professionals and clerical workers 10 or 15 years hence); some of the issues, problems and opportunities facing universities; and what several universities are doing to respond to this situation.**

The imminent penetration of information technology and equipment into the office, the home, the school, and other social and economic institutions has been widely discussed, predicted, and (occasionally) denounced during the last few years. All of us who have any contact with the data processing and information services industries are well aware of the situation and are familiar with numerous articles dealing with it in a variety of ways. However, an administrator or executive who is responsible for budget planning for an educational institution or government agency cannot use the visions of a futurist as a basis for planning and decisions. Unfortunately, neither can a person with such responsibilities ignore this type of technology and the cost advantages of using it in making plans for the future.

This situation poses a paradox for long-range planning in such organizations; the problem is a more complex one for large institutions and bureaucracies, for which the planning, financing, and management problems associated with adopting technology effectively are more formidable than those faced by smaller institutions. It would be easy to sit back and observe how and to what extent society adopts information systems technology -- and when this becomes clear -- to base institutional planning on the knowledge and experience of others. The costs of doing this -- in terms of potential savings and in terms of educational programs needed by many of our students -- may be too great to permit us the luxury of sitting back and waiting to see how the future works out.

About a year ago, I began to spend substantial amounts of time on two projects:

- Word processing -- bringing it into our own office and planning and coordinating its development within the entire University: and
- Replacement of our telephone system (currently based on a central office No. 5 crossbar switch).

A comprehensive study of work processing in a university environment, released in preliminary form last summer by the Stanford University Center for Information Technology, was a seminal source document in identifying some of the implications and trade-offs in office automation technology as applied to a university. Our background information on telephone systems, the independent telephone interconnect industry, and the role of "supercontroller" exchange switches, and local data networks in information technology applications has been derived from the trade press; from conversations with persons at other universities who are facing similar problems: and from numerous meetings with manufacturers, vendors, and consultants. Several of these meetings have been on a "non-disclosure" basis, making it difficult to cite references for some of the information on which our thinking has been based.

The deeper we got into these projects, the more complex, far- reaching, and interconnected the two became, leading eventually to the conclusion that the University of Minnesota had to begin some serious, comprehensive, integrated planning for the role that information technology will play in its operations and programs during the next ten or fifteen years. Some of the specific factors that persuaded me of this are as follows:

1. The Federal Communications Commission has deregulated the telephone/telecommunications industry (as of March 1980), opening it to a degree of competition and innovation previously unknown in this country.

2. Many large and powerful corporations are investing very substantial amounts of money in all aspects of the microelectronics/computer/dataprocessing/telecommunications industries. They are also jockeying for position with various regulatory agencies and the courts. In addition to the expected participants in this action (IBM, Xerox, AT&T), the participants include firms such as Exxon, ARCO, Citicorp, and Volkswagenwerke.

3. The U.S. Postal Service is aggressively trying to begin experiments and development in the electronic delivery of mail -- both by facsimile and by direct transmission of machine- readable texts produced with word processing equipment. Their efforts to do this are being vigorously opposed by commercial interests who wish to preserve this market for themselves and to restrict the Postal Service to its traditional role.

4. The French Postal, Telephone, and Telegraph Agency has embarked on a plan to place the telephone directories throughout France on-line by 1990, replacing the printed telephone book with on- line data display devices built into every telephone instrument. The integrated telephone-data-display devices will be manufacxured by the French telecommunications industry, with a design goal of $100 to $250 unit cost. The first 250,000 units are being installed in western France in 1981. This program has been developed with the expectation that the savings from eliminating telephone books and information operators, plus the income from a wide range of additional services once the display devices are installed in every home and business, will more than pay for the more expensive terminal equipment. The plan has the full support of the French government, which also expects a thriving export market to develop.

5. Several large U.S. universities have taken major steps toward planning and/or implementing a future in which all or major portions of their student bodies, faculty, and staff will have ready access (in or near their offices ) to data terminals or microcomputers connected to a variety of centralized and distributed information and computing resources. These institutions include Stanford and Carnegie- Mellon (reports on these two have appeared in the Chronicle of Higher Education in September 1980 and January 1981, respectively), Michigan, and Chicago.

Of these developments, the French plan to have a combination telephone/keyboard/display terminal in most homes and offices by 1990 is perhaps the most compelling. When and if this happens, the wherewithal will be in hand for the "office of the future" and the many home services that have been discussed in theory and tried as experiments. We are all aware that the technology for these developments has been in hand for many years. The if, when, and how of office automation and home information services depends on economic factors (the economies of mass production and the availability of investment capital to support manufacturing and distribution on a mass scale); and on political factors (regulation, tax policies, legislation and regulations to protect existing industries, etc.). If this technology becomes ubiquitous in the rest of society, then education will use it, too, even if enrollments are dropping and budgets are being reduced, retrenched, and rescinded.

What will the future look like in our institutions ten or fifteen years hence if this comes to pass? I will assume that most of you are familiar with some of the common descriptions of office automation and the future effects of technology on education. The sections following merely summarize, in list form, my own views on some of the more significant and likely characteristics of technology in a university environment.

### The "Office of the Future:"

- A desk on which rests a combination voice/data work station.
- A chair for the occupant of the office.
- A chair or two for visitors, infrequently occupied because most business is transacted through the work station.
- Electronic message and mail systems for most internal communications; these will connect to local and national electronic mail systems. (This will be the fate of the U.S. Postal Service and the Campus Mail.)
- Access to institutional, local, and national data banks and information utilities.
- There may be in and out baskets for residual paper documents, though there will be few of these; if there are in and out baskets, a waste basket will also be present.
- A book shelf containing professional and reference books and journals. (I do not expect that much of this kind of information will be

on line in data banks or stored on video disks within the next ten or fifteen years.)

- Not much else; in particular, no file cabinet.
- Most faculty and students at most of the leading research universities will own, have custody of, or have ready access to a terminal or communicating microcomputer.

### Some characteristics of educational programs in the future:

Here I am addressing primarily the higher education level with which I am most familiar. This list is "conservatively speculative." I have omitted those applications that I believe are unrealistic as early as 10 or 15 years from now. I believe that some of the developments listed will come to pass within this time span, but not all of them. Similarly, there will be developments not listed which come into use, either because I have overlooked them or because I have underestimated the future. In some cases, higher education will lead the way, and in other cases it will surely lag behind other levels of education and certain other segments of society.

- Most quantitative subjects (science, mathematics, economics, accounting) will rely substantially on computation to teach concepts and applications.

- Quantitative methods and computer technology will play a significant (but perhaps not dominant) role in many heretofore relatively non-quantitative subjects (the softer social sciences, education, literary and language analysis).

- Computer-assisted instruction will be playing a major role in certain kinds of instruction -- e.g., languages, elementary music theory, problem-solving in many disciplines.

- Word processing will be used widely as a tool by faculty members and students in

  * vocational programs such as journalism and secretarial science;
  * general English composition and writing programs;
  * all subjects which require substantial amounts of creative or expository writing.

- Computer graphics will be used in a substantial way for instruction in design (engineering, exterior, interior, artistic), business, and perhaps other disciplines.

- In many universities, bibliographic information from the institution's libraries and a few outside bibliographic utilities will be available directly to most or all faculty members and students.

- Administrative and management information for institutional operations will routinely be available and used in machine-readable form rather than on paper; much of it will be distributed electronically.

- Some (but not all) institutional and interpersonal communications will occur via internal and external electronic message, mail, and conferencing systems.

### Basic components of a future university information system:

- Computer devices of all sizes (from micro- to super-) which will manage and manipulate

  * text;
  * graphic images of reasonable complexity;
  * quantitative data.

  Some of these devices will be located in individual offices serving one or a few persons; some will serve portions of the campus; some will serve the entire campus and perhaps beyond; and some computing resources will be obtained from outside organizations.

- Digital voice systems, including voice store-and-forward capability.

- Electronic telecommunications systems linking computer devices and digital voice terminals into a common network, including electronic message and mail systems; capability to transmit voice and data at rates of 64 kilobits/sec or higher; and access to a variety of external local, regional, and national telecommunications networks.

- Data bases to assist in the processes of teaching and learning, research, service, and institutional operations.

### Alternative architectures of a future university information system:

The architecture of an information system needs to reflect the organization it serves in order to facilitate its management and its services. Although I have not begun to understand this important aspect of information systems designs to my own satisfation, there seem to be two main types of architecture from which very large organizations can select:

- A "supercontroller" network; a star configuration integrating data and voice communications, with a sophisticated computerized exchange switch at the center of the star to route and control the communications from a wide range of voice and data terminals and computers.

- A bus network carrying data in packets, using broadcast technology and collision detection schemes; plus an independent telephone (voice) network (star configuration). Occasional gateways between the voice and data networks provide for access from one to the other.

The opportunities facing education as this future looms on the horizon are three:

- Improvement in the management and operations of our institutions.

- Control of costs via increases in productivity of employees and replacement of labor with technology where appropriate; this will be required of us, not to increase our profits, but rather to survive budget reductions and avoid pricing ourselves out of business.

- Improvement of educational programs -- keeping them current with the expectations of society and the job market our students enter.

When will the time be ripe for universities and other educational institutions to begin introducing information systems technology so that almost all employees, faculty members, and students use it regularly? A realistic answer must be: as soon as it is cost justified in terms of operations, management, and/or educational needs -- and no sooner. This leads to another question: what will it take for this cost justification to be realized? Two factors will be necessary — which is the chicken and which is the egg I will leave for you to decide:

- Widespread use in business and industry or other large segments of society; a bandwagon effect of which education, unfortunately, cannot afford to be the vanguard, but from which we will also be unable to remain aloof.

- Substantial reductions in the cost of equipment through mass production and distribution; one more round of cost reduction similar to the microcomputer revolution of 1978-1980 will be needed for most of us.

Indications are now beginning to appear that the economy is moving in these directions. In addition to some of the specific facts cited above (numbered items 1 through 5), two general trends can be identified:

- A market of the order of hundreds of billions of dollars has been identified by a number of large corporations -- they are aiming their strategic planning at this market, even though they are reluctant to talk about their plans in specific detail. In our "non-disclosure" discussions with several of these corporations, however, it has been clear that they are keeping one eye trained over their shoulder so that they can change course if the response of the market falls far enough short of their expectations.

- Information workers (office, white collar) make up such a large fraction of the U. S. labor force (far outnumbering industrial and agricultural workers) that, without the application of information systems technology to control costs and manage paperwork and record- keeping, our economy and society are in BIG trouble.

Given this somewhat ambiguous situation -- clear indications of the directions society and the economy are going, but evidence that is a little mixed and not yet conclusive -- what should large educational institutions be doing by way of long-range and strategic planning? If I had the answer to that question, I would probably be speaking to you as a high-paid consultant rather than as a professional colleague wrestling with many of the same problems you are. Here are some of the things I think that universities ought to be doing, and that we are trying to do at the University of Minnesota:

- Begin to establish a method and a group of people to carry out integrated, strategic planning for information systems, starting at an exploratory level and moving into specifics as soon as circumstances warrant it.

- Watch the penetration of information systems technology into business and industry carefully, including prices and cost-saving potentials, so that circumstances can be identified quickly in which it becomes reasonable to move into detailed planning.

- Begin using various pieces of information systems technology as soon as they are cost justified; for instance, departmental accounting systems, advanced voice/data telecommunications exchanges, etc.

- Begin experimenting with word processing as a tool for writing as soon as possible -- to improve institutional operations, to reduce the costs of document preparation, and to learn how educational programs (from general composition courses to vocational progams in areas such as secretarial science and journalism) need to be modified to take advantage of these new tools.

- It would appear that, for many institutions,there is or soon will be an opportunity to achieve substantial cost savings in telephone services at the same time that a variety of data communications, store-and-forward, and call- and message-transfer capabilities are provided. These additional capabilities can serve as a foundation for first-generation electronic message and mail systems and office automation. The potential benefits of state-of-the-art and state-of-the-market telephone/telecommunications exchange systems are so great that the opportunity should certainly be explored if it presents itself or if it can be caused to present itself.

- Finally, there is the controlling factor -- financing. How can informations systems services and facilities be financed at the same time that we are coping with various kinds of resource reductions and limits? This will take expertise and a great deal of imaginative and creative thinking. If my expectatiions for the future are valid, then it is not a matter of whether we can obtain access to these kinds of information services -- it is a matter of when and how. We cannot meet our responsibilities to our students and to the public if we do not find a way. My own expertise in the area of financing is almost non-existent. Two of the issues I forsee, however, are as follows:

  * Should information services and facilities be viewed as a recurring operating cost or as a capital cost? Different institutions may arrive at different answers to this question, depending on how they are financed and on the structure of their budget. Whatever the answer, it should be recognized that we are dealing with a recurring cost of doing business. Each item of equipment acquired will have a finite useful lifetime, after which it will be too expensive to continue using it rather than replacing it with more recent and more powerful equipment. This may mean establishing sinking or revolving funds (an anathema in some public educational institutions), or developing some kind of understanding amongst state government officials as to how such matters should be managed.

  * Educational applications of information technology depend on initiaves from faculty and staff members (e.g., introduction of computer- assisted instruction and library automation). When these initiatives come, there must be some understanding on the part of institutional administrators of how they can be sustained financially as a regular part of the instructional or support program. In many institutions, this issue may reduce to one of funding through departmental budgets or through the central administration. Again, the answers will differ among institutions, as they should. The important thing is that, when an initiative does come from a faculty member or a department, there should be a timely and clear administrative response to it. This will require creative and imaginative thinking to identify ways of funding such initiatives -- when they are well-thought-out and justified. The goal should be to provide positive incentives and rewards for such initiatives rather than negative incentives in the form of administrative paralysis and non-response.

At the University of Minnesota, we are in the initial stages of planning; that is, we are planning for planning. We are not in a position to take initiatives similar to those of Carnegie- Mellon and Stanford Universities because we do not yet know what we should be doing in the context of our own programs, opportunities, and constraints; and because we are facing the possibility of budget reductions and adjustments which will affect those opportunities and constraints. We are in a position, however, to know that, within the next few years, we will have to replace our telephone system. There are two reasons for this:

- Our current telephone exchange switch system is nearing the limit of its capacity: it lacks many features currently available on newer systems; it provides some features through key telephones and other special attachments at considerable expense; and it is increasing in cost to us despite the lack of these features.

- There is a real opportunity to replace our current voice system with a digital telecommunications exchange system that will accomodate both voice and data messages and provide a foundation on which to build more extensive information services -- and at the same time to hold down or actually reduce costs. We cannot afford NOT to explore such an opportunity with great care.

If nothing else, the necessity of planning for replacement of our telephone system is forcing us to begin planning more broadly for information services.

Whether any of our experiences at the University of Minnesota are relevant or transferable to smaller educational institutions or to other levels of education, I am not sure. I am sure that there are many important differences related to size, mission, level of education, and source of funding (public or private). I believe that there are important lessons to be learned by all of us in sharing information on our respective experiences and in understanding our differences with regard to information services planning. And I would welcome a vigorous and continuing discussion among us as a useful aid to developing this understanding and to helping us all respond to the opportunities and challenges that we face during the next several years.

# DISTRIBUTED SYSTEMS IN THE SCHOOL DISTRICT ENVIRONMENT

James R. Settlemire
Manager, Data Base/Systems Software
Minnesota School Districts Data Processing Joint Board
T.I.E.S.
St. Paul, MN

The improvements in modern technology are beginning to remove the technical and economic constraints which have prevented computing resources from moving away from the central computer facility to the grassroots personnel of the school district. This paper explores the following questions relating to the distribution of computing resources:

1. What is a distributed system?
2. What are the motivating forces behind distributing systems?
3. What is the planning process for distributed systems?
4. What are the potential problems with distributed systems?

## Introduction:

### Background and History

In the early 1970's school districts, faced with growing demands to automate administrative processes, turned to central data processing centers to meet those needs. At the time this was a cost justified alternative for the development of data processing systems designed to meet the unique needs of education at the school district level. The development process for the more progressive data centers included information systems supported by integrated data base management systems and accessed via on-line remote terminals.

Now as we move into the decade of the 80's, the economic and technical constraints have changed in the areas of data communication, vendor software, mini/micro technology and costs. What was once not feasible may be possible with today's technology. There are now the means to return to the remote user the control of their work functions that were centralized a decade ago.

### What is a distributed system?

The definition of distributed systems (alias: distributed processing, distributed computing, decentralized data management, partitioned processing, dispersed processing, network systems, and remote data services) is similar to trying to define the Management Information Systems of the 1970's. It is defined within the framework of one's own organization, its progress within the state of the art, its current and future needs, and its long range plan and objectives.

There are more than enough definitions of distributed systems available. A few selected ones are as follows:

"Distributed processing is the deployment of computerized data processing functions where they can be performed most effectively, at costs usually less than those of other options, through the electronic interconnection of computers and terminals, arranged in a network adapted to the user's characteristics."

"The use of multiple autonomous, but cooperating, computer processors arranged to support common system goals."

"An integrated collection of systems that harmonize to serve one corporate entity and are either controlled centrally or given a degree of local autonomy to achieve the planned organizational goals."

Rather than present another definition of distributed systems perhaps it would be helpful to look at what is potentially distributed:

1. Costs
2. Staff
3. Responsibility and control (of data, processing, scheduling, priority setting, software/hardware, and problem solving)
4. Complexity

5. Technical support
6. Hardware (introduction of standalone mini/micro computer systems in the remote user environment)
7. Software (computer operating software and application software)
8. Security (transfers new levels of security to the remote user environment)
9. Data processing supplies (purchasing, logistics of remote site inventories)

Distributed systems can be thought of as a conceptual design approach that will transfer some part of the above components away from the centralized location to various locations and levels of the remote user environment.

### What are the motivating forces behind distributed systems?

1. The need for a positive organizational plan to meet the problem of the emerging micro computer controlled devices appearing in the organization. Without a plan the organization will find itself in the midst of a multitude of disjoint and incompatible hardware/software devices that users want to "hook up" to the information system.

2. As the level of experienced user knowledge and self confidence increases, the demand for more sophisticated processing at the grass roots level of the organization rises.

3. Remote users want work function independence from the central computer, with more control and scheduling over critical processes.

4. To provide the flexibility to expand the system by small modular units of hardware/software instead of by quatum upgrades of mainframe hardware.

5. As central data bases become large, complex, and slower the remote users want local access to portions of their data.

6. Some transaction volumes reach a point where it is no longer feasible to process all of them at one central point.

7. The remote users want intra-organizational data transfer (i.e., school to administration office, school to school).

8. The remote users want to implement applications that are not available from the central site.

9. The remote users want more control over their data processing costs to achieve their own priority goals.

10. Increasing hardware/software capabilities at decreasing costs.

11. Upper management is seeking more management information through simulation, projection, and forecasting with the results displayed locally at crucial decision points (i.e., board meetings, salary negotiations, etc.)

12. Remote users have differing needs and goals because of size, population change, and educational services delivered.

### The planning process for distributed systems

° Organizational planning

1. The long range plan of the central organization should be in place with the overall strategy specifying how distributed systems will be implemented.

2. A program of user education should be carried out at various levels in the organization so that users have an understanding of the basic concepts of distributed systems. This is important in order to gain acceptance of the implementation plan and to provide the knowledge for later selection of the various alternatives that will be offered.

3. Develop a set of policies, rules and procedures that outline the district and the central data centers responsibility for the implementation and support of distributed systems.

Central data center planning

1. There are two distinct design and development efforts involved in the distributed system.

    A. The general framework of the hardware/software network that transfers and manipulates data in the information system.

    B. The specific applications (payroll, mark reporting, scheduling, etc.) which are designed to operate within the distributed system network available.

2. Various levels of conceptual configurations of hardware and software capabilities should be developed based on remote processing needs defined in the users needs assessment.

3. Present the proposed conceptual configuration options to the remote user community to determine if it meets their remote processing needs. Use practical examples to show how each configuration can be used to meet specific needs. Ask the user groups to choose the configurations that best meet their anticipated requirements. From the assembled reponses identify the configurations that best meet the user needs and fit well into the organizations goals and constraints.

4. Develop the selected configurations into specific vendor hardware/software alternatives that will be available in the distributed system. Present the options to each user including the hardware, soft-

ware, and cost specifications requesting a response for intent of participation.

5. Design the overall distributed system using data indicated by user intent and level of implementation.

    A. Design the central site systems that are needed to support the distributed system (i.e., remote file transfer, remote printing, remote data entry, remote job requesting, telecommunications network, etc.).

    B. Design the remote site systems that interact with the central site systems.

    C. Design the specific application systems that are to be implemented in the distributed system.

6. Install each type of remote hardware/software system for development and testing at the central site.

7. Select an appropriate number of remote users for a period of pilot testing.

8. Begin a phased implementation of users ready for installation and production.

- Remote user planning

1. Assess the existing data processing capabilities.

    A. Staff assignments and responsibilities.
    B. Applications already implemented.
    C. Applications planned for future implementation.
    D. Other outside data processing services.
    E. Describe (narrative or diagram) current work flow of data processing related applications implemented. Define points where distributed systems would help meet distributed system objectives.
    F. Inventory present data processing equipment, their placement, and applications processed.

2. Rationale for implementation of distributed systems.

    A. Develop district objectives and identify requested or demonstrated needs. Examples:

        Reduce turnaround time for edit/error lists.
        Off-line data entry.
        Implement some standalone applications.
        Store some data remotely.

    B. Define specific applications to be implemented. Examples:

        Print mark reporting edit at the high schools.
        Standalone inventory system at the warehouse.
        Implement financial transactions with remote data entry.
        Prepare administrative personnel report from remote data.

    C. Based upon the list of applications to be implemented chose the appropriate distributed system hardware/software configurations supported by the central data center. Selection should include the appropriate device, size, capabilities, and placement of the equipment (i.e., administration office, secondary schools, etc.).

    D. Areas for consideration

        Weigh computing power/speed against people cost (i.e., it may be more cost efficient to use faster equipment versus high people cost for longer or extra hours).

        The hardware/software decision should be made after the needs assessment, applications to be implemented, and processing capabilities are defined.

        Review the district responsibilities to support the distributed system in the remote environment.

        Define the local district requirements for off loading data from the remote computer for backup and history.

        Review intra-district security considerations (i.e., who can access what data from what locations?).

        Is the plan to use centralized or decentralized intra-district data processing? (i.e., perform all work at the administration office or move some functions out to the school level.)

        Considerations involving physical space, phone connections, noise level, and power requirements for the placement of equipment.

        Long Range Planning:

            What are the future needs for more devices?
            What are the future data storage needs?
            What are the future printing needs?
            What are the plans to implement standalone processing?

3. Develop a comprehensive data processing plan for the district.

   A. Cost projections by budget year
      - Hardware
      - Software
      - Communications
      - Data processing supplies
      - Site preparation
      - Staffing
      - Training

   B. Implementation plan
      - Hardware/software installation by location with timelines.
      - Application system installation by system with timelines.
      - Staffing and training plan with timelines.

Distributed System Problem Areas

1. The established, tightly controlled, central data center efficiency may be lost as the information system is extended out into the user environment.

2. The extension of the application systems deeper into the remote user environment encounters new demands for local uniqueness and customizing.

3. Requirements for well defined policies/procedures to protect the integrity of the remote system from unauthorized hardware/software changes.

4. The control and coordination of software to all distribution points with central coordination of problems with vendors.

5. Methods and staff resources for central technical support of remote user problems.

6. The more extensive and sophisticated the design of the distributed system the greater the requirements for a complex central system to control the network.

7. Coordination and control of centralized and remote copies of the organizations data.

8. Complexity of hardware/software exceeds user capability, willingness or acceptance.

9. The existing operational problems occurring at the central site will be distributed in some measure to the remote site.

10. The new distributed concept must be retrofitted into a large, complex, and entrenched system already in place (applies to both system software and applications software).

11. Some application systems may need to be rewritten to truly distribute their functions.

12. Requirements for accurate estimates of data to be transferred within the network.

13. The avoidance of premature obsolesence of vendor hardware/software in the long range plan.

14. Providing configurations that are realistic, acceptable, and meet the needs of a wide range of users.

15. The delivery of remote processing capabilities greater than the ability to implement application systems which results in idle equipment.

16. The adjustment at the central site to the impact of the distributed network.

17. Not all remote users will want to implement some form of distributed systems and will have to be supported centrally.

18. User expections that central site costs will decrease immediately to off set the increased remote site cost increases.

Conclusions:

While the implementation of distributed systems is relatively new, and the technology is changing at a rapid pace, the experience of the past is still valuable in addressing the problems. The professional approaches of good planning, design, and implementation are still valid and clearly imperative due to the new level of complexity. The introduction of distributed systems is an evolutionary process which will require careful planning as more sophisticated processing is placed in the hands of new levels of users. If it is not handled properly the result will be another era of user frustration and complaint with the data processing experience. If it is handled properly, it will place state of the art technology in the hands of remote users which will enable them to meet their management information needs effectively.

# A SYSTEMATIC PROCEDURE FOR PROBLEM ANALYSIS AND PROPOSAL DEVELOPMENT

Esther R. Sinofsky
Frederick G. Knirk
Department of Instructional Technology
University of Southern California
WPH 801  P.O. Box 77963
Los Angeles, California 90007

As money becomes tighter, the need to seek outside funding increases. To compete for these funds, the ability to identify the specific need(s) and to develop winning proposals is of prime importance. Making effective decisions becomes crucial. Thus, the purpose of this presentation is to bring together several educational systems management tools which can reduce the time and effort spent in identifying goals and developing proposals. It will introduce ways to systematically increase a proposal's quality through better identification and writing of objectives, reduction of typist-related errors, and facility of editing additions, revisions, and modifications.

If you are going to write a proposal, make your time and energy count. It seems as though we all have too little time to do the many things which we value and would like to do. On the national average nine out of ten proposals which are developed and submitted for funding are in fact exercises in frustration - they are not funded.

Your proposal should ooze quality. The old adage, "If it's worth doing, do it right, " is certainly appropriate when one is attemping to fund a better "mousetrap". When examining the how, what, why, who, where, and when issues it soon becomes obvious that the two creative activities reflecting the what and how are difficult to develop into technology and to teach. What your mousetrap is going to cover initially must be decided by yourself. You may get assistance from your immediate environment which has identified problems or from your professional interests which reflect on unresolved questions. How you are going to develop it may be only vaguely clear at this point. On the issue of Why it becomes possible to use the technology of "needs assessment." Needs Assessment primarily answer these three questions:

a) What is the status quo?
b) What should be the status quo?
c) What discrepancies, if any, exist between what should be and what is?

The discripancy between what is and what should be is in fact a definition of your project activity. The data for this needs assessment may come from an examination of an existing problem environment or from brain storming activities with other individuals who are relatively expert on the problem or on relevant techniques for solving similar problems.

In order to examine the Why of the future and to examine the relevance of this project to the future, the "Delphi" technique may be used. In fact, the Delphi technique is not as good a future projection of technique as it is a procedure for obtaining group consensus on the relative value, need, and method of a common project. The Delphi technique is written up in some detail by e.g., Olov Helmer. In essence, this procedure involves having a panel of experts respond to the need, time, and tasks to be performed. The panel members do not interact with each other so that peer pressure or personality problems do not influence the decision making structure. This anonymity also makes it possible for individuals to easily change their positions as additional data is presented, without loss of face.

Having generally examined the How, What and Why, it is possible now to go back and refine the How. Task Analysis can be performed so that the individual activities or events to-be-performed in the study may be examined in some detail. It is possible at this point to show the relationship of various decisions and activities on each other as they are reflected in flow chart. The data for this procedure may be obtained through examining similar activities or, again, by using a panel of experts who can instruct one on the probable future activities related to the project.

The What may be refined by translating the task statement into the Project's Behavioral Objectives. The task statements reflect behavioral statements of activities or events to be performed. In order to refine a task statement into a behavioral objective, it is necessary to specify in detail the audience and behavior to consider and specify under what conditions the task or behavior is to be performed, and to consider and specify the degree to which the behavior must be done. A format which we have found to be particularly productive involves the ABCD method of writing objectives. Using this method, the audience statement is first identified followed by the behavioral statement, after which the conditions and degree statement follow. An example of an ABCD developed objective would be "the students in the third grade reading class" (the audience statement) "will be able to spell 30 words" (the behavioral statement) "without the use of their textbook or dictionary" ( the condition statement) "with at least eighty percent accuracy" (the degree statement). (The audience statement may be written only once if the objectives are all appropriate to the same audience). The advantage of the ABCD method is that it establishes the criteria for determining a project's success and thus serves as a basis for comparing programs. This is often critical when several proposals are competing for the same funds.

At this time, it is desirable to bring together the various statements which have been written out in rough form by the various individuals or committees working on the proposal. As a guide for developing the outline for the proposal, I suggest that you consider How to Prepare a Research Proposal, by Krathwohl, which is available in the second edition from the Syracuse University bookstore. Phi Delta Kappa's fastback #143, The Process of Grant Proposal Development, is another useful proposal-writing guide. At this point, the activities or events to be accomplished may have to be massaged somewhat to reflect the request for proposal (RFP) or the "Foundation Directory" stated interests of the philanthropic foundation to whom the proposal is to be sent.

As a computer-oriented audience, you will certainly be using the computer as a word processor. Screen editors, such as FINE and/or editors, such as TECO and run-off formaters, will substancially increase the quality of your proposal , and decrease the costs fo developing a proposal due to decreasing the personnel time requirements. As the initial task statements were developed on the word processor, it is very easy to simply expand these task statements into objectives, by recalling the existing behavior statements (task statements), and adding the audience condition and degree portions of them. This will reduce development time and confusion. Having written the task statements, you may wish to use them later when personnel requirements are estimated, and when the cost for the project is estimated. It is a waste of time and effort to retype and restate these activities more than once.

If a large timeshare system is used to develop the proposal, it is possible for the different members working on the proposal to access the same file and rapidly develop the document. Because the system is relatively error free, a correctly thoughtout and typed document need never be subjected to the secretarial pool for the inclusion of additional typing errors.

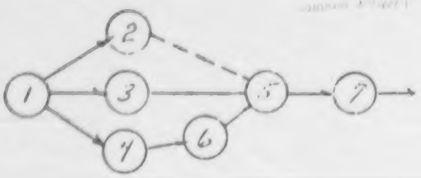
The who, where, when, and how much? issues can now be examined. The use of a "PERT network" will help the proposal developers to estimate who will be working on the project and how long it will take them. Using a relatively easy formula for estimating the time requirements for the project

$$TE = \frac{A + 4M + B}{6}$$

Where TE= elapsed time
M= most likely time
A= optomistic time
B= pesimistic time

not only will result in a project estimate of time which is good as can be developed, but it will also force you to estimate or determine who will be working on the project. The total elapsed time for the project can be determined by examining the longest path through the PERT network (see Figure 1),

Events:
1) Begin Project
2) Hire Secretary
3) Project Eval Hired
4) Data Spec Hired
5) Project Tasks Listed
6) Obtain Data Processor
7) Project Work Plan
8) Etc.



can be examined to determine the critical path. The critical path is defined as that path through the network which requires the greatest amount of time to get from the beginning to the end of the network. This critical path, then, reflects the total elapsed time required to perform the project.

Now that you have estimated who will be working on the project, and how long it will take to perform the project, it is possible to determine where the project will be conducted, and how long the project will take. The use of the PERT cost chart (see Figure 2), will help in this matter.

Estimating Cost

| Event | Person 1 | Person 2 | Computer |
|---|---|---|---|
| A | 2 days | 1 day | 2 hours |
| B | 3 days | 4 days | 1 hour |
| C | --- | 5 days | 2 hours |
| Total | 5 days | 10 days | 5 hours |
| | x salary/day | x salary/day | x $/hour |
| | Cost | Cost | Cost |

The time required per person for each of the activities or events listed is multiplied by the unit salary, and then totaled. Any additional supplies or expenditures required per event, such as computer time, are also included. A summation of these costs is then the total project cost. While this procedure is not fool proof, it is certainly better than the wild guesses which are often made when preparing a cost estimate for a proposal. It is important that the cost be realistic. Do not understate your development costs, unless you can afford to subsidize the completion of the project yourself. It is also necessary to be sure you do not overestimate the cost of the project for a competing proposal, with a much lower cost estimate, will probably recieve the contract instead.

In summary, the use of the techniques discussed in this paper may result in the facilitating your developing and writing a proposal. These techniques should help in guiding you over what can easily be a frustrating experience.

# DELTA COMPUTER-MANAGED INSTRUCTION SYSTEM

Robert L. Uffelman
Edward E. Boas

University of Delaware
Newark, Delaware 19711

ABSTRACT: DELTA Computer-Managed Instruction uses three programs, EXPLORE, REPORT and CREATE, to manage delivery of instructional modules that may include terminal performance objective, pre-check, learning packages with enabling objectives, post-checks, remedial feedback and alternate post-checks. The system is written in BASIC Plus for Digital Equipment Corporation PDP 11 series with a RSTS/E operating system and printer terminals.

The DELTA CMI system was developed to manage the instruction of a multioccupation career exploration program for secondary school students. Frantz and Boas (April 1977) report that initial acceptance by these students, teachers and administrators resulted in computer program modifications that permit several users to access the files simultaneously. This improved utilization according to Frantz, and Boas (July 1977). The comparative effectiveness of computer-managed instruction, printed modules and teacher-lecture was compared by measuring student achievement of stated criterion referenced performance objectives. The findings of Boas (1979) indicate that student achievement was highest with those students who received computer-managed instruction . The DELTA CMI system was used for staff development at a four-campus technical and community college. The capability for quick editorial revision, delivery to widely distributed sites and flexibility in scheduling utilization led to development of positive attitudes among instructors toward implementation of computer-managed instruction for their students as reported by Frantz, Boas and Matthews (1979). Additional computer program enhancements were added to facilitate the delivery of remedial instruction, easier access by users and improved security of record-keeping components.

Instructors, with assistance from a computer system coordinator, utilize the DELTA CMI system to manage and deliver individualized instructional modules with three programs: EXPLORE, REPORT and CREATE. These programs are written in BASIC Plus and run on a Digital Equipment Corporation PDP 11/50 and 11/70 using RSTS/E operating systems. The BASIC run-time system must include MAT and PRINT-USING statements. It does not require STRING arithmetic, LOG or TRIG functions. A run time system generation in this manner, without a floating-point processor, but with a four-precision mathematics package will take up 16K of memory, leaving 16K for the user job space. With a floating-point processor, the run-time system will undoubtedly take less memory. DELTA CMI requires 16K of memory to run. The peripheral storage requirements for the instructional modules and student records will vary according to the files used. The modules are delivered to students via printer terminals and include: a terminal performance objective; a pre-check; packages containing enabling objective(s), instructional resources, learning activities and feedback; post-checks; remedial feedback; and alternate post-checks.

Personnel involved with implementing a system will include instructors, the computer system coordinator and students.

Steps for initiating the system are:

1. Instructors prepare modules in the recommended format.
2. The computer system coordinator loads the three programs on the computer and establishes accounts for classes.
3. Instructors or data clerks enter the modules into temporary files and add student names to the class accounts.
4. The computer coordinator merges the temporary files into the master data base.
5. Students obtain assignments from a computer terminal after a briefing by the instructor.

When the system has been initiated, instructors can monitor class progress by obtaining reports for each student, each instructional module or summaries of class activities. Class lists may be modified at any time by the instructor. Modules may be taken off-line for revision as temporary files and then reloaded by the computer system coordinator.

The EXPLORE program delivers the modules to the student who signs on to the system and is provided a list of the clusters that are available to him. Then areas in the cluster are displayed. The student selects an area in which to work and is given a list of modules. The list briefly describes the terminal objective for each module. The student may also print out the complete objective for any module. The student is then asked if he wishes to take a pre-check. If so, it is printed at the terminal, the student is logged of and works on the pre-check. After completing the pre-check, the student returns to the terminal, accesses the program and inputs his answers. His next activity is prescribed by the EXPLOR program according to the instructor's plan. If the student elects not to take a pre-check, all the learning packages in the module are printed at the terminal and the student is logged off to work on the module. Next, he signs on to a terminal for a post-check. After completing the post-check off line, he enters his answers by logging in again.

The system informs the student if he has achieved the stated criterion for terminal performance objective of a module. If he was not successful, remedial instruction is prescribed, and he takes another post-check after completing this assignment. Additional post-checks and remedial instruction may be provided by the instructor's plan. Progress is recorded by the REPORT program.

The REPORT program is the instructor's gradebook and keeps current records for all students using EXPLORE for the class. It records what module is being used, time and date the student started working on the module and time he last worked on it. The REPORT program also prints out a detailed listing of student answers to the pre-check and post-checks. This feature and an item analysis of incorrect answers enable the instructor to add or delete students from those using EXPLORE. They also allow the instructor to specify what students are qualified to take alternate post-checks and to receive remedial feedback.

CREATE is the tool of the computer system coordinators. It is used for merging the modules properly for use by the EXPLORE program. The coordinator can also specify which modules are on-line or off-line at any given time. CREATE lets him take a module off-line for up-dating without interfering with use by others. The coordinator also can specify which of the modules can be accessed by specific students. Another option of the CREATE program allows students to by-pass all of the module except for a post-check. In this way, the system can be used for delivery of an examination or competency measure for certifying qualifications of students.

The DELTA CMI system is very efficient for students, instructors and system coordinators. Its format and operation make it possible to adapt almost all types of instruction for use on the computer system. Using TECO, a simple text editing program, instructors can enter their own modules or adapt others. The system coordinator can merge them into a master data base for use by students in several classes. In this way, a large and varied selection of modules can be made available in many subject areas.

Computer-managed instruction facilitates development of mastery learning or performance-based curriculums. It provides instructors with accurate records of student progress and more time for individual and small-group instruction. By developing banks of instructional modules, institutions or school districts can standardize terminal performance objectives while encouraging individual instructors to use a variety of resources for teaching their own students. Maintenance of basic skills can be assured by periodic rechecking and providing for any necessary remedial instruction.

## REFERENCES

Boas, E. E. An analysis of instructional delivery systems in vocational education comparing computer-managed instruction , teacher delivered module (paper) and lecture-demonstration for the same instructional module (Doctoral dissertation, Temple University, 1979).

*Dissertation Abstracts International*, 1979, *40*, 2624A. (University Microfilms No. 7924043, 146)

Frantz, N. R., & Boas, E. E. A computer managed instructional system for a multioccupational approach to vocational-technical education. *THE Journal Technological Horizons in Education*, April 1977, *4*(4), 24-26.

Frantz, N. R., & Boas, E. E. *The development and field testing of a computer-managed delivery system in multioccupational programs for vocational education* (Project No. 498AH60022). Washington, D. C.: U. S. Department of Health, Education, and Welfare, July 1977.

Frantz, N. R., Boas, E. E., & Matthews, J. I. Using computer managed instruction for staff development at a technical and community college. *THE Journal Technological Horizons in Education*, March 1979, *6*(3), 30-31

# ALASKA ON-LINE RESOURCE INFORMATION BANK

Ralph Van Dusseldorp
Professor of Education
University of Alaska, Anchorage
3221 Providence Drive
Anchorage, Alaska 99504

In order to provide schools information on instructional resource available, a computer network was established. The network involves a central computer at the Department of Education and a microcomputer in each district. Information about instructional resources available - people, agencies, materials and programs - are stored in the central computer. Each resource is classified according to several ERIC - like descriptors. When a school needs a particular type of instructional resource, descriptors (for example: elementary, arithemetic, gifted) are entered into the microcomputer. These are then transmitted to the central computer which selects the resources that match the descriptors, and transmits information about those resources to the microcomputer at the school.

## Introduction

In order to appreciate the significance of the use of computers by schools to aid in obtaining information about educational resources, one must first understand the unique educational problems of Alaska. Many of the problems have to do with the remoteness of schools and school districts. Alaska is, of course, our largest state. The following facts provide some indication of its size:

- It is more than twice as large as texas.
- It encompasses four time zones.
- There are 52 school districts in Alaska.
    - One district is 1,000 miles long.
    - One district is larger in area than all the New England states combined. That district has ten schools and 1,200 students.
- If Alaska were superimposed on the continental United States, one end would be in Florida while the other end would be in the state of Washington.

But it is not only the size of the state that causes unique educational problems, it is the remoteness of schools, the difficulty of communications, and the cost of operations. Some districts spend over $20,000 per student per year for education. It costs about $1,000 for round trip air fare from a remote district to the state capitol. Many schools are very small and remote. In some cases it is several hundred miles to the next school or the district office. Some schools are in villages without a telephone. Some district offices are in villages with only a single phone serving the entire village.

The remoteness of the schools causes many problems. The one we are concerned with here is being out of the mainstream of educational thought and the lack of information about current educational practices and resources. Professional libraries in remote schools are very limited. Opportunities for teachers and administrators to attend conferences and conventions are also limited. It is difficult for remote schools to obtain information about sources of help for program improvement. If, for example, a remote school wishes to develop a program in environmental education, it is very difficult for the school to find materials in that area or people who are knowledgeable in that area.

Thus, when we began the development of the computer system our goal was to assist all Alaska schools, but especially the remote schools, in gaining information about educational resources. By educational resources we mean information about persons, agencies, programs or materials that can assist a school in the updating or improvement of instruction.

Before our computer project was started, an attempt was made toward making educational resource information more available through the establishment of the Alaska Talent Bank. In the establishment of the Talent Bank, a number of persons with expertise in particular areas who of education and were willing to conduct workshops in Alaska schools or provide consulting for schools were identified. Information about those persons was organized and printed in a catalog which was distributed to schools. This catalog helped schools identify resource persons who could provide help in instructional improvement but did not satisfy all the needs for resource information. In addition, the process of updating the catalogs was cumbersome.

Thus, a system was needed that would provide ready access to information about a wide variety of educational resources and which could be easily updated. At about that time the Alaska Department of Education was in the process of installing a computer network to provide an electronic mail system (EMS) for Alaska schools. That network involved the installation of a microcomputer in each of the 52 Alaska school districts as well as the regional resource centers and some other educational agencies. A larger central computer was installed at the Department of Education, and communication among the computers was through the Alaska telecommunications system. The same hardware and communication system was used for the educational resource information system that is the subject of this paper.

## The Knowledge Base System

The system developed was written in DEC BASIC PLUS. It operates on a DEC 11/70 which is the central computer for the EMS system. Access to resource information stored in the computer is through GNAT microcomputers which are located in all school districts, regional resource centers and some other agencies, and which serve as terminals and provide direct access to the data base.

The data base, called a knowledge base in this application, consists of abstracts of information about available educational resources. The knowledge base currently includes about 800 abstracts. The abstracts are divided into the following categories:

Service Agencies. Public agencies that provide materials or services to Alaskan schools.

Talent Bank. Alaska residents with particular skills who are willing to provide assistance to schools.

Promising Practices. Programs developed in Alaska school districts which have been evaluated and designated as being exemplary.

Alaska-developed Materials. Materials developed by Alaska schools to meet particular needs of Alaska.

Nationally Validated Programs. Exemplary/model programs validated by the USOE and NIE and selected as appropriate for Alaska.

Commercial Resources. Private individuals, agencies or companies that sell materials or services to Alaska schools.

Possible resources are screened by personnel in the Alaska State Department of Education (DOE). When a resource is judged to be appropriate for inclusion in the knowledge base, an abstract for that resource is developed. An example of a knowledge base abstract is shown in figure 1. In order to provide for selective search and retrieval of abstracts, descriptors are entered with each abstract. The descriptors are words or short phrases that describe the resource. A standard set of descriptors is used. The set of descriptors was developed by DOE personnel and comprise a subset of the ERIC descriptors.

Search and retrieval of abstracts is primarily through the use of descriptors and an AND-OR-NOT operator logic. For example, entry of

```
          READING
    AND   COMPREHENSION
    AND   ELEMENTARY EDUCATION
     OR   SECONDARY EDUCATION
    NOT   REMEDIAL INSTRUCTION
```

will result in retrieval of abstracts having the descriptors READING and COMPREHENSION and either ELEMENTARY EDUCATION or SECONDARY EDUCATION but will exclude those with REMEDIAL INSTRUCTION as a descriptor. The user also selects the amount of information from the abstracts that is to be displayed or printed. For example, the user may wish to just print titles or may wish to print complete abstracts. In addition to retrieving abstracts according to descriptors, users can retrieve abstracts by type (service agency, talent bank, etc.) or can retrieve single specific abstracts by entering the identification numbers. An example of user interaction with the system is designating a search for knowledge base abstracts according to descriptors is shown in figure 2.

The knowledge base system was implemented in in three stages. Initially, all retrieval was done by DOE personnel in response to requests. Then personnel in the regional resource centers were trained on the system and they provided retrieval services for local schools. Finally, at least one person in each school district was trained in the system and, when that training was complete, all school districts had direct access to the knowledge base. Thus, currently all 52 Alaska school districts, the four regional resource centers, and certain other educational agencies have direct access to the knowledge base.

## Figure 1

### Example of a Knowledge Base Abstract

```
IDEN NO.       : TB000001
TYPE           : TALENT BANK
NAME/TITLE     : JOHNSON, MARILYN KAY (SPECIAL EDUCATION)
REGION         : SC
ADDRESS        : UNIVERSITY OF ALASKA
                 ANCHORAGE, AK
TELEPHONE      : (907) 465-2841
CONTACT PERSON : TALENT BANK COORDINATOR/AK DEPT. OF ED.
DATE           :
FEE:$125.00 PER DAY HONORARIUM PLUS MATERIALS COST (NOT COVERED BY TALENT BANK)

DESCRIPTION:
MARILYN JOHNSON OFFERS TO ADMINISTRATORS AND TEACHERS EXPERTISE ON
LEARNING DISABILITIES, RECOGNITION, IDENTIFICATION AND REMEDIATION,
GENERAL SPECIAL EDUCATION AND MAINSTREAMING.  HER PROGRAM IS FOR
THE ELEMENTARY LEVEL PRIMARILY BUT SHE ALSO HAS EXPERIENCE WORKING WITH
HANDICAPPED ADULT PROGRAMS.  DEPENDING ON THE TASK SHE MAY WORK WITH
GROUPS OF FIVE TO ONE HUNDRED.  SHE HAS EXPERIENCE IN ELEMENTARY
TEACHING, SPECIAL EDUCATION TEACHING AND UNIVERSITY TEACHER TRAINING
EXPERIENCE.  HER EDUCATION BACKGROUND INCLUDES A BACHELOR'S DEGREE IN
ELEMENTARY AND SPECIAL EDUCATION AND MENTAL RETARDATION, A MASTER'S IN
MULTIPLE HANDICAPPED AND DOCTORATE IN LEARNING DISABILITIES.  THE
LENGTH OF HER PROGRAM IS DEPENDANT ON NEED BUT FOR TWO DAYS MINIMUM.
REFERENCES INCLUDE MARSHA BUCK, DIRECTOR OF SPECIAL EDUCATION.  FOR THE
KETCHIKAN GATEWAY SCHOOL DISTRICT, AND JIM RICH , DIRECTOR OF SPECIAL
EDUCATION FOR THE ANCHORAGE BOROUGH SCHOOL DISTRICT.
()
DESCRIPTORS:  ADULT EDUCATION                  LEARNING DISABILITIES
              ELEMENTARY EDUCATION             SPECIAL EDUCATION
              HANDICAPPED STUDENTS             TEACHER EDUCATION
              INSERVICE EDUCATION
```

### Use Statistics

Statistics on use of the knowledge base are kept automatically by the computer system. Those statistics reveal that during the period of approximately one year that the system has been completely implemented, there were 2,281 uses of the system. Thus, the average number of uses per district was about 40. The frequency of use varied considerably from district to district. Eight districts did not use the system at all. At the other end of the scale, one district used the system 229 times.

A number of other statistics that provide information for maintaining the system are kept. For example, use statistics show that the 2,281 uses resulted in the selection and listing of 21,736 abstracts, or an average of about ten abstracts per search. This would seem to indicate that most searches result in the selection of an appropriate number of abstracts.

Use statistics is also kept on the number of times each descriptor is used and the number of times each abstract is retrieved. The data on the number of times each descriptor is used serves two purposes. Descriptors which are infrequently used are examined to determine if they are appropriate descriptors and if they should be changed or eliminated. That data also indicates the type of information users are looking for. Infrequently retrieved abstracts are examined to determine if the descriptors for those abstracts are appropriate or if those abstracts should be deleted from the system.

### The Hardware and Software

As mentioned earlier, the knowledge base operates on a DEC 11/70 central computer with GNAT microcomputers serving as terminals in the schools. The programs for the systems are written in DEC BASIC PLUS. The system takes 48 K to run and the storage of the 800 abstracts currently in the system takes about 3,000,000 bytes of disk storage. The system is always operated in an interactive mode, both for file maintenance and for retrieval of information.

### Summary

The resource information system has been quite successful. It has provided users with a convenient way of obtaining information about educational resources and has provided a convenient method of keeping files updated. Periodically, all the abstracts in the files are printed and sent to the appropriate persons and agencies. Corrections and changes are made and the files are updated. In addition, the files are updated as the DOE personnel become aware of changes.

**Figure 2**

**EXAMPLE OF KNOWLEDGE BASE SEARCH INSTRUCTIONS**

```
                      INFORMATION RETRIEVAL
                      ---------------------

RUN BY (E=EXIT)                             ? DOE

RUN FOR (E=EXIT)                            ? DOE

SEARCHER'S NAME                             ? RALPH VAN DUSSELDORP

SELECT BY (D=DESCRIPTOR, I=ID #, E=EXIT)? D

INDICATE WHICH TYPES OF ABSTRACTS TO SEARCH (Y=YES, N=NO)

          SERVICE AGENCIES                       ? Y
          TALENT BANK                            ? N
          PROMISING PRACTICES                    ? Y
          NATIONALLY VALIDATED PROGRAMS          ? Y
          ALASKA-DEVELOPED MATERIALS             ? N
          CLIENT PROFILES                        ? N
          COMMERCIAL RESOURCES                   ? N

ENTER DESCRIPTORS

AND:  ? READING
          OR:  ?

AND:  ? COMPREHENSION
          OR:  ?

AND:  ? ELEMENTARY EDUCATION
          OR:  ? SECONDARY EDUCATION
          OR:  ?

AND:  ?


NOT:  ? REMEDIAL INSTRUCTION
NOT:  ?

IDENTIFIER (<RETURN> FOR NONE)  ?

CONTACT PERSON (<RETURN> FOR NONE)  ?
REGION (<RETURN> FOR NONE)  ?


INDICATE WHICH ITEMS ARE TO BE PRINTED (A AFTER ID #=ALL, Y=YES, N=NO):

IDEN NO.                    ? Y
TYPE                        ? Y
NAME/TITLE                  ? Y
ADDRESS                     ? Y
TELEPHONE                   ? Y
CONTACT PERSON              ? Y
DATE                        ? Y
COST/FEE                    ? Y
DESCRIPTION                 ? N
IDENTIFIERS                 ? N
DESCRIPTORS                 ? N
REGION                      ? N
SEARCH SUMMARY              ? N

READY PRINTER AND <RETURN> (E=EXIT)
```

# COMPUTER SCIENCE SUMMER SCHOOL OFFERING

## A PLACE TO START

Edward E. Wachtel, Chairperson

Computer Education Steering Committee

Shaker Heights City School District

15600 Parkland Drive

Shaker Heights, Ohio 44120

The Shaker Heights City Schools offered a non-credit junior high computer science course in the summer of 1980 for a limited number of students as a pilot for computer science offerings to be developed during the 1980-81 school year.

The course included thorough instruction in the BASIC programming language on a variety of hardware, two field trips and discussions ranging over a variety of computer related topics.

The structuring of the course and its eventual execution and evaluation gave valuable information for the further development of credit computer science offerings in the Shaker schools. It has also spurred the possible creation of future summer computer science offerings at all levels and similar offerings open to adults through the local recreation board.

### STRUCTURE

Since the course was an initial step into this curriculum area it was felt necessary to limit the number of participants. The basis for this limit was the amount of hardware available for use in the district. Since only eight machines (four terminals attached to an H-P 3000, two TRS-80's, two Apple II Plus's) were available, enrollment was limited to sixteen students, i.e., two persons per machine.

A regular summer school program exists in the Shaker Heights City Schools so the usual student fees and teacher salary were used for the course.

Field trips were conducted using parent volunteer drivers.

Students were solicited from those who had finished sixth, seventh or eighth grade in June. The class was filled on a first come first served basis yet was a very satisfactory mix with respect to sex, age, race and ability.

### MATERIALS

No standard text was used for the course. Instead, a selection of acquired bits and pieces were used. This selection consisted primarily of a BASIC programming overview provided by the teacher, Computer Conversations[1], and the CAMP[2] materials. While this material was somewhat lacking with regard to the history of the computer and the fundamentals of how a computer operates, it was adequate for learning BASIC programming. More materials are becoming available and are being acquired to prepare a more thorough course for future use.[3,4,5,6] It is felt that the use of a standard text would limit flexibility in teaching about a rapidly developing field and should be reserved for higher level courses with more specific goals and limited scope.

### DAILY OPERATION

While many students entered the course with varying programming skills, most were totally lacking in global computer knowledge and none could be considered expert programmers. It became important to teach a great deal of general background about the computer and to begin the programming instruction from a least common denominator of the participants. This latter was especially important in that all participants, even those with a fair programming knowledge needed a firm foundation upon which to build good programming habits.

To that end, programs were required of all participants from the overview and the other materials that traced a spiralling level of programming knowledge and skill. Vocabulary lists were kept on available board space with new words added as encountered.

Those who became very competent programmers were provided with additional challenge by being asked to alter existing game or graphics programs in some innovative way or to create new programs of this type. The level of programs relating to mathematical or scientific topics became somewhat limited late in the course due to the level from which the students were selected. It was felt that students with some algebra experience learned BASIC more readily and were able to do more with it than those who had had limited algebra experience.

Two field trips gave a "real world" perspective to the course. A trip to the international telecommunications and data processing center of a major corporation presented the participants with a view of the use of large computers in the everyday business world. A trip to a local personal computer store provided a glimpse of small business computing, word processing, and home use of the microcomputer.

## EVALUATION

An evaluation questionnaire was circulated at the end of the course. Questions were loosely based on the Mathematics Teacher overview of computer literacy.[7] While the results may not have had great statistical significance they did indicate a generally positive reaction by the students to the course and what they learned. Some issue opinion questions were included primarily out of curiousity regarding student reactions to discussions of those issues.

COMPUTER SCIENCE - SUMMER 1980 RESULTS AVERAGED

RANK EACH STATEMENT FROM 1 (STRONGLY AGREE) TO 10 (STRONGLY DISAGREE)

A. I HAVE LEARNED A GREAD DEAL IN THIS COURSE 2.7
COMMENTS:

B. I HAVE ENJOYED THIS COURSE. 2.16
COMMENTS:

C. I UNDERSTAND STRING VARIABLES. 4.12
COMMENTS:

D. I UNDERSTAND HOW TO USE SOME GRAPHICS ON THE APPLE OR TRS-80 2.30
COMMENTS:

E. I FEEL I AM A COMPETENT BASIC PROGRAMMER. 2.43
COMMENTS:

F. I HAVE A BETTER UNDERSTANDING OF THE HARDWARE AND SOFTWARE MAKING UP A TOTAL COMPUTER SYSTEM. 2.34
COMMENTS:

G. I FEEL COMFORTABLE USING A COMPUTER. 1.98
COMMENTS:

H. THERE ARE MORE ADVANTAGES TO USING THE COMPUTER. 2.32
COMMENTS:

I. COMPUTER SCINECE SHOULD BE REQUIRED OF ALL STUDENTS AT SOME TIME IN THEIR SCHOOLING. 3.43
COMMENTS:

J. I UNDERSTAND BETTER THE EMERGING ROLE OF THE COMPUTER IN TODAY'S SOCIETY. 2.03
COMMENTS:

K. I THINK A NATIONAL COMPUTER INFORMATION BANK IS A GOOD IDEA. 2.49
COMMENTS:

L. COMPUTERS INCREASE UNEMPLOYMENT. 2.86
COMMENTS:

M. THE USE OF COMPUTERS TO PROCESS PERSONAL INFORMATION SHOULD BE CAREFULLY REGULATED. 2.13
COMMENTS:

N. COMPUTERS ARE HIGHLY ADVANTAGEOUS TO SOCIETY. 3.10
COMMENTS:

## CONCLUSIONS

The course did in fact serve as a very useful pilot for ensuing credit offerings. A one-half credit course is being offered at the high school this year based largely on the curriculum from this summer course. One-half credit electives in computer science will be offered for ninth graders at each of the two junior highs in the district during the 1981-82 school year again based largely on this same curriculum. It is hoped that future summer offerings will be on a credit basis.

While a ratio of two participants per machine provided for excellent service, certainly more could be accommodated with proper scheduling of machine and non-machine oriented tasks. An optimum number probably should be established to maximize usage without jeopardizing the experience.

Those students who participated in the summer course have functioned through this school year as a nucleus around which many other interested students tend to cluster, have served as able assistants in existing computer clubs and in one case as an advisor to an elementary school computer club.

While the summer school structure in Shaker Heights provided for the assessing of fees and the payment for instruction, it is felt that in a situation where this structure may not exist some fee should be attached for participation in a course of this type. This would help separate those with a genuine interest from those with only idle curiosity. Further, this would offset some of the expenses and perhaps assist in providing fair remuneration for the teacher assuming the responsibility for such a course.

## REFERENCES

1. Computer Conversations (Student cards and Teacher Guide), Earl Orf and Diana Hestwood, The Math Group, Inc., Minneapolis, MN, 1974

2. Computer Assisted Mathematics Program (CAMP) LaFrenz, Johnson, Hatfield, Katzman, Kieren, Walther; Scott, Foresman and Company,1969

3. Problems for Computer Solution, Stephen J. Rogowski, Creative Computing Press, Morristown, N.J., 1980

4. Computers: From Pebbles to Programs, The Skills Group of the Center for Humanities, Inc., Mount Kisco, N.Y. (Media Kit)

5. BASIC: An Introduction to Computer Programming, The Skills Group of The Center for Humanities, Inc., Mount Kisco, N.Y. (Media Kit)

6. Adventures of the Mind, A Series On Personal Computing, ITV Co-op, Falls Church, VA, 1980 (A Television Series)

7. "Computer Lieteracy - What Is It?", Johnson, Anderson, Hansen, Klassen, Mathematics Teacher, February, 1980

# A FIVE YEAR TIME LINE
# FOR THE INTEGRATION OF COMPUTERS
# INTO THE SHAKER HEIGHTS CITY SCHOOLS

Edward E. Wachtel, Chairman

Computer Education Steering Committee

Shaker Heights City Schools

15600 Parkland Boulevard

Shaker Heights, Ohio 44120

Committee Members:

Clair Rankin
Madeline Obrock
Evelyn Burns
John Bazeley

## INTRODUCTION AND PHILOSOPHY

"Although computers are indispensible to science, business and government, they have not realized their full potential in American education. In an information society, a computer literate populace is important, and we may go through a crisis trying to develop it." [1]

"For today's children, understanding computer fundamentals is one very important factor in building an informed citizenry for the future. Students who are computer literate will have better career opportunities and less career shock. They will be better able to cope in a world with rapidly moving and ever-changing technology. Thus, in this era called the Age of Information, schools can no longer delay in bringing the computer into their curricula."[2]

"What we are doing is creating a new class structure around wealth .... this time the wealth of information. Like today's 'haves and have-nots' we will be a society of 'knows and know-nots'." [3]

The 70's were the decade of computer technology and the 80's are destined to become the decade of computer education. Thus it is important to the growth and effectiveness of the Shaker Heights City Schools to plan for the 80's an educational thrust that includes the development of students and staff who are aware of and can contend with the overall social, economic and educational implications of the computer. It is the belief of the Computer Education Steering Committee (CESC) that a five year plan must be developed to insure that this school system will continue to make available to all students enrolled the most relevant education possible, and thus transmit those items deemed necessary for success in an increasingly technological and computer oriented society.

## LONG RANGE GOALS

### I. STUDENTS[4]

A. At the completion of elementary school the average student will have:

1. demonstrated an awareness of the importance of the computer within society
2. demonstrated necessary minimum skills in using hardware, e.g., typing, manipulation, use of turnkey programs.
3. demonstrated enhanced learning by use of C.A.I. and C.M.I. programs.
4. evidenced improved learning through use of tutorial and remedial programs available during and after school hours.
5. demonstrated positive attitudes, values, and higher motivation to

learn generally as a result of the use of the computer.

B. At the completion of junior high school the average student will have:

   1. demonstrated enhanced learning through the use of C.A.I. and C.M.I. programs.

   2. demonstrated a moderate level of computer related skills - "BASIC" vocabulary, debugging and revising algorithms.

   3. written original programs

C. At the completion of senior high school those students who have enrolled in computer science electives will have:

   1. demonstrated ability to use a computer in home financial management. (checking accounts, savings, budgeting, cost analysis)

   2. written original programs in several disciplines. (Enrollees in Computer Science I & II.

   3. demonstrated abilities necessary to participate in advanced computer science. (Enrollees in Computer Science III, IV, & V)

II. STAFF [5]

A. Staff members who have participated in the basic in-service programs will have:

   1. demonstrated the integration into their repertoire of teaching skills some use of the computer in the classroom, e.g., drill, remediation, simulation, C.A.I., C.M.I.

   2. demonstrated the ability to perform simple programming.

   3. communicated the impact of computers on society to their students as it relates to that teacher's subject area.

B. Staff members who have participated in the Staff Computer Science courses will have:

   1. demonstrated a comprehensive level of computer literacy.

   2. demonstrated advanced programming skills

   3. developed software available for district use.

EVALUATION AND RESEARCH

Because of the rapidly changing technology in hardware and software, the recommendations of the Time Line, based on present technology, are subject to change. Therefore, evaluation procedures should be structured for use by DIPC, administration and staff for continual assessment of the effectiveness, use and suitability of hardware, software and staff.

An ongoing research tool must accompany this evaluation procedure to determine the effect of the computer on learning, attitudes and values of both students and staff throughout the district.

COMPUTER SCIENCE OFFERINGS

I. STUDENT - If each secondary schools acquires a computer laboratory, a directed system of course availability in this area was felt to be a most beneficial opportunity. While all students would acquire an initial background and familiarity with the equipment, it was felt that the additional courses listed below would permit rigorous student learning in this area.

Course offerings listed below should be discussed with secondary math and science teachers since CESC recommends the formation of a separate Computer Science department.

The summaries below are not meant to be thorough course descriptions but only comments as to the general nature the course structures might take.

A. 7th and 8th Grade Survey - Provides all Students with a basic computer literacy[4] including a limited introduction to programming in one computer language, e.g., BASIC.

B. Computer Science I - Expansion of basic computer literacy from 7th and 8th Survey courses with emphasis on more sophisticated programming skills in the one language introduced.

C. Computer Science II - Advanced programming for the seriously interested student. Student use of the computer for projects in other curriculum areas. Introduction of simulation, interactive and word processing programming skills. Wider exposure to the HP - 3000 system.

D. Computer Science III, A,B,C, etc. - Instruction in languages other than the one selected for Computer Science I & II, e.g., FORTRAN, COBOL, PASCAL.

E. Computer Science IV - Advanced computing, e.g., Numerical Analysis for Computers, Statistical Applications for Computer, Advanced Word Processing.

F. Computer Science V - Opportunities for major independent study projects involving the computer.

II. STAFF - If the school system is to undertake positive support of this program, then in-service programs and Computer Science course offerings should be made available to the staff as noted in the Staff Training section of the Time Line. A brief description of the Staff Computer Science offerings follows:

The CESC recommends that the school system finance these course offerings unless individual participants desire credit for possible advancement on the salary schedule and/or further state certification.

A. Staff Computer Science I - An introduction to computers, their uses, capabilities and limitations. Fundamental programming skills in one language. Survey of educational applications.

B. Staff Computer Science II - Provides participants with a thorough level of computer literacy with more sophisticated programming skills.

C. Staff Computer Science III - Use of programming skills learned in courses I and II to develop CAI and CMI, simulation and other educational uses of the computer.

ESTIMATED FINANCES

These figures represent at best only a "Ball Park" estimate and are subject to revision dependent upon changes in cost and nature of hardware, commercial software, staff salaries, etc.

| | | |
|---|---|---|
| Staff | $626,000 | (5 years) |
| Hardware | $418,800 | (5 years) |
| In house systems materials | $ 50,000 | (5 years) |
| Teacher Training | $ 50,000 | (5 years) |
| CESC | $ 7,000 | (1 year) |
| Summer Writing | $250,000 | (5 years) - Should be partially absorbed in existing summer writing budget. |
| Summer Materials | $ 50,000 | (5 years) |
| System Wide CAI - CMI | $100,000 | (5 years) |
| Total | $1,551,000 | |

The CESC feels that the budget for the district computer education program should be independent of present district budgeting to prevent losing necessary funds for normal department operations.

POSSIBLE PROBLEMS

The CESC anticipates three major problems in the implementation of the recommendations on the Time Line.

A. Cost - computer hardware and software are expensive. Skilled programmers and knowledgeable professionals are in short supply and command high salaries.

B. Staff Acceptance - The majority of adults are not knowledgeable about computers. While our staff is talented it needs the patient opportunity for adaption. Implementing a district wide computer education program would involve major investments of time for additional training.

C. Scheduling - In an already crowded curriculum, it is always difficult to schedule something new. In the first semester, building principals and guidance counselors should be cognizant of computer courses starting the second semester of the school year and the associated scheduling and released time requirements for these courses and their preparation.

ADDITIONAL COMMENTS

DIPC - The District Instructional Program Committee should assume a major role in the integration and evaluation of computers in the educational process.

Data Processing - Vocational Education - The CESU has considered the issue of Data Processing - Vocational Education and notes that the current structure of our school system is not oriented in a vocational direction. However, we recognize the growing interest, need and demand for this type of program. We feel it could be most beneficial for the Shaker Heights City Schools to investigate the establishment of a multi-district center for this purpose in the very near future. Funding from various sources could and should have a peripheral value in the support of our own academic program.

Elementary Typing Skills - While a fully developed typing course would not be necessary or advisable beginning in the elementary years, the increased use of the computer at that level may require the teaching of at least rudimentary knowledge of the keyboard and its efficient utilization via the "touch" system.

CONCLUSION

Using computer technology to the optimum level should be the goal of the Shaker Heights City Schools. Computers used to achieve traditional goals within present modes of operation would perpetuate the mundane.

Computers skillfully integrated into subject area curricula along with creatively designed computer science courses would assure that students and staff would become familiar with this important technology. To undertake this effort would continue and enhance this school district's resolve to make available to its students the most relevant education possible.

|  |  | 80-81 |  | 81-82 |  | 82-83 |  | 83-84 |  | 84-85 |  |
|---|---|---|---|---|---|---|---|---|---|---|---|
|  | Basic | Add per school | Totals | Add per school | Totals | Add per school | Totals | Add per school | Totals | Add per school | Totals |
| HARDWARE ELEM. | 0 H.P.'s<br>12 Micros | 0 H.P. terminals<br>1 disk drive for micros | 0 H.P.'s<br>12 Micros | 1 H.P. terminal<br>4 Micros | 9 H.P.'s<br>48 Micros | 0 H.P. terminals<br>19 Micros with 3 disk drives | 9 H.P.'s<br>138 Micros | 0 H.P. terminals<br>10 Micros with 3 disk drives | 9 H.P.'s<br>228 Micros | 0 H.P. terminals<br>5 Micros | 9 H.P.'s<br>273 Micros |
| JR. HIGH | 6 H.P.'s<br>10 Micros | 0 H.P. terminals<br>3 Micros with [illegible] | 6 H.P.'s<br>16 Micros | 0 H.P. terminals<br>6 Micros | 6 H.P.'s<br>18 Micros | 0 H.P. terminals<br>10 Micros | 6 H.P.'s<br>48 Micros | 0 H.P. terminals<br>10 Micros | 6 H.P.'s<br>48 Micros | 0 H.P. terminals<br>0 Micros | 6 H.P.'s<br>68 Micros |
| SR. HIGH | 4 H.P.'s<br>0 Micros | 10 H.P. terminals<br>6 Micros | 14 H.P.'s<br>6 Micros | 16 H.P. terminals<br>7 Micros with disk drives | 30 H.P.'s<br>13 Micros | 0 H.P. terminals<br>7 Micros with disk drives | 30 H.P.'s<br>20 Micros | 0 H.P. terminals<br>7 Micros | 30 H.P.'s<br>27 Micros | 0 H.P. terminals<br>0 Micros | 30 H.P.'s<br>27 Micros |
| STAFF |  | Add during this year | Total Staff | Add during this year | Total Staff | Add during this year | Total Staff | Add during this year | Total Staff | Add during this year | Total Staff |
| CENTRAL |  | 1 Director of Educational Computing (by 6/30/81) | 1 | 1 Educational Programmer<br>1 Clerical | 3 | 1 Troubleshooter (by 6/30/83) | 4 | 1 Educational Programmer<br>1 Clerical | 6 | No additional | 6<br>1 Director<br>2 Programmers<br>1 Troubleshooter<br>2 Clerical |
| BUILDING |  | 2 Computer Science Teachers (by 6/30/81) | 2 | 2 Computer Science Teachers<br>1 Elementary Computer Specialist | 5 | No additional | 5 | No additional | 5 | No additional | 5<br>4 Comp.Sci.Teachers<br>1 Elem.Comp.Spec. |
| COMPUTER SCIENCE CURRICULUM ELEM. |  | Computer Clubs | Summer '81<br>Course writing of Comp.Sci. I,II,III<br>Initiate interim Computer Science Department at the Secondary level | Computer Clubs | Summer '82<br>Course writing of Comp.Sci. IV and 7th & 8th Survey | Computer Clubs | Summer '83<br>Course writ<br>Computer Sc<br>Computer Sc Department structured operational | Computer Clubs | Summer '84<br>Update and revise Comp. Sci. courses, if necessary | Computer Clubs | Summer '85<br>Update and revise Comp. Sci. courses if necessary |
| JR. HIGH |  | Computer Clubs |  | Computer Clubs<br>9th Grade Comp. Sci. I |  | Computer Clubs<br>9th Grade Comp. Sci. I<br>7th & 8th Grade Survey |  | Computer Clubs<br>9th Computer Science I<br>7th & 8th Survey |  | Computer Clubs<br>9th Computer Science I<br>7th & 8th Survey |  |
| SR. HIGH |  | Computer Science I & II (from existing format) |  | Computer Science I,II & III |  | Computer Science I,II,III, IV & V |  | Computer Science I - V |  | Computer Science I - V |  |
| [illegible] |  | - Opening Day Professional Development - Elementary and Secondary Workshops<br>- 1st Semester - Building Mtgs., Admin. Coun., [illegible], Elem. Prin. presentations by Computer Educ. Steering Committee (CESC)<br>- 2nd Semester - H.P. demos at the Sr. High (Biorhythms, games, etc.), Staff Computer Science I is offered | Summer '81<br>Make available to teachers any info on workshops, mtgs, Univ. courses, etc dealing with computers<br>Organize Opening Day activities and 1st Semester In-Service mtgs. | - Opening Day, Prof. Dev. - Keynote speaker on Computers and technology in education, reports on summer curriculum and software groups by levels<br>- 1st Semester - In-Service on basic hardware, rudimentary CAI/CMI and available material<br>- Both Semesters - Staff Computer Science I & II | Summer '82<br>Encourage summer study on computers.<br>Structure further Opening Day and In-Service activity | Same as 1981 - 1982 with the addition of Staff Computer Science III | Summer '83 | Continue In-Service and Staff Computer Science I,II and III as need warrants |  | Continue In-Service and Staff Computer Science I,II and III as need warrants |  |
| ( C.A.I. / C.M.I. ) |  | CESC - Investigate and preview available commercial materials for all levels<br>[illegible], Administration and staff determine structure and recruit staff for Summer Writing by April 2, 1981 | Summer '81<br>Summer software groups (K-6,7-9, 10-12) for selecting materials, writing,etc. to determine pilot projects for next school year | Test 1 or 2 pilot projects in various subject areas at each level (K-6,7-9,10-12)<br>Sept. 1 - Format ready<br>1st Sem. - Prepare for use<br>2nd Sem. - Pilot functional | Summer '82<br>Summer software groups as in '81 but in new subject areas<br>Evaluate '81-'82 pilots | Continuation of successful '81-'82 pilots with exposure to staff affected by these programs<br>Affected staff makes decision on adoption of '81-'82 pilots into full use by April 2, 1983<br>Sept.1 - Format for new pilots ready<br>1st Sem. - Prepare new pilots for use<br>2nd Sem. - New pilots in use | Summer '83<br>Summer software groups and evaluation continue in the same pattern as '81 through '8 | Piloting process continues as before<br>Institute in district the approved '81-'82 pilots | Summer '84<br>Summer software groups continue as before | Successful and approved pilots continue to be adopted in the district.<br>Continue evaluation and update of existing materials<br>Continue evaluation of new materials for possible piloting and adoption |  |
| [illegible] |  | - Organize Staff Computer Science I to begin 2nd Sem.<br>- Funds allocated to begin upgrading professional and building libraries re: computers (books, periodicals, AV, etc.) | Summer '81<br>Organize Staff Computer Science II to begin 1st Sem.<br>Begin generating evaluation procedures | 1st Sem. - Released time by substitutes for preparation of pilots<br>Director of Educational Computing takes charge and CESC becomes advisory<br>Continue library fund allocation | Summer '82<br>Organize Staff Computer Science III<br>Begin systematic evaluation of program | Continue support as in previous years | Summer '83<br>Major evaluation of equipment, materials, progr and staff for a GO - NO GO decision | Initiate centralized computer curriculum library facility and software distribution system ( in-house users group) | Continue evaluation process |  |  |

Where possible CESC encourages the hiring of teachers and administrators with interest or expertise in computers.

FILMED FROM BEST COPY AVAILABLE

## REFERENCES

1. Billings, Karen, in Microcomputing, June 1980, Pg. 100

2. Hopper, Judith, from a presentation at the National Educational Computing Conference, Norfolk, Va., June 23-25, 1980

3. Schwartz, Peter, quoted in "And Man Created the Chip", Newsweek, June 30, 1980, Pg. 51

4. Johnson, Anderson, Hansen, Klassen, "Computer Literacy - What Is It?", Mathematics Teacher, February 1980, Pgs. 91-96.

5. Taylor, Poirot, Powell, "Computing Competencies for School Teachers" Proceedings of NECC/2, Pgs. 130-136

# WORD PROCESSING EVALUATION AND SELECTION

Russell E. Weitz
Director, Design & Development
Minnesota School Districts Data Processing Joint Board
T.I.E.S.
St. Paul, MN

The paperwork world has descended upon us and we look for faster and easier ways of processing reams of paper. How can a person determine if a word processing machine can be used in an office environment? Can it reduce the amount of secretarial work load and at the same time increase productivity? These questions will be addressed and hopefully answered.

The initial step is to determine if there is a need. This can be done by conducting a study to determine the type and quality of typewritten communication produced at the secretarial stations. Each typist should be given a series of code sheets to complete for each document typed. Information to be recorded for each document should include the number of pages, category, generation, input modes, confidentiality and turnaround. The study should not extend beyond two weeks or a ten work day period.

A study conducted at TIES indicated there were four categories of typing:

Correspondence - Letters, memorandums, notes, etc.

Text - Reports, speeches, manuals, directories, minutes, etc.

Statistical/Forms - Numerical and columnar work and completion of pre-printed forms.

Repetative - Repetative typing could be done in a pre-recorded fashion (e.g., same letter to several people, "boilerplate" paragraphs, document assembly, workshop announcements).

The study further broke down each category by original vs. revised material. Below is detailed the number of original and revised lines produced in each of the four categories.

| Category | | Total Lines | Word Processor Per Hour | | Total Hours |
|---|---|---|---|---|---|
| Correspondence | - Orig | 4,219 | 135 | = | 31.2 |
| | - Rev | 803 | 400 | = | 2.0 |
| Text | - Orig | 7,428 | 135 | = | 55.0 |
| | - Rev | 6,588 | 400 | = | 16.4 |
| Statistical/ Forms | - Orig | 1,466 | 80 | = | 18.3 |
| | - Rev | 46 | 355 | = | .1 |
| Repetative | - Orig | 1,080 | 553 | = | 1.9 |
| | - Rev | 1,321 | 553 | = | 2.3 |
| | Estimated Total Hours | | | | 127.2 |

The actual number of hours spent on typing during the test period was 171.6 hours. Comparing the actual hours with the estimated total hours through the use of word processing resulted in a time savings of 44.6 hours over a ten day period.

Now that the need has been identified, the real task lies ahead to determine what vendor or type or word processor should be used. There are some key concepts and characteristics to decide upon which will help to narrow the field of eligible vendors to a reasonable number.

## Stand Alone Vs. Shared Logic

The stand alone word processor would have some kind of visual display, a keyboard, storage media, hard copy printout, and the computer that ties everything together and performs the necessary editing, file management and input/output functions.

The shared logic word processor has the same properties as the stand alone but has additional input and/or output devices located in other areas.

## Hardwired Vs. Soft or Programmable

The hardwired word processor exists, but currently is a minor portion of the industry. The editing and operation functions are rigid and can only be changed through engineering or a change of machine. The main advantage with this device is you get exactly what you see.

The soft or programmable word processors make up the biggest share of the market place. The editing and operating functions exist as written commands to the computer. This type of device provides the user with flexibility to change and incorporate new releases of software that could be of benefit to the organization and at the same time extend the useful life cycle of the device.

## No Display Vs. Limited Display Vs. Cathode Ray Tube

The no display device has a light emitting diode (LED) displaying anywhere from sixteen to thirty two characters. The operator enters information just with a typewriter. The most recent characters typed are displayed and the characters do not have to be printed on hard copy. This provides visual verification and is also used by the designers as a prompting tool and/or communication vehicle between the operator and operating system.

The Cathode Ray Tube (CRT) display is the most expensive and provides expanded communication features for the operator and operating system. Most CRT's have a twenty four line capacity therefore a full page of 8½ by 11 data cannot be displayed at one time. There are some vendors that provide a full screen able to display the complete letter size document.

## Diskette Vs. Cassette

Most word processors use the eight inch or five and a quarter inch flexible or floppy disk. The exception to this would be the larger shared logic word processors. Disk storage provides indexed access to information which decreases the amount of time to retrieve information. It also eases the workload of the operator by having the operating system keep track of the exact location of information on the file.

The magnetic cassette tape is a less expensive device but all information is stored in sequence and the location of data is a task of the operator. There are vendors that have designed tape drives that operate very fast thereby cutting down the amount of time it takes to seek needed data.

## Operator Ease

The techniques to operate a word processor vary by the number of vendors in the market place. Review the operator instruction manual and determine how easy it is to operate and the number of codes the operator must learn. Vendors that do not provide good manuals will indicate something to the prospective buyer. Also inquire as to the amount of time it takes to train an inexperienced operator.

The display device may be a critical feature in your quest for word processing. Display devices are not the same, therefore prospective buyers should understand various features and concepts associated with word processors.

## Raster Scan Vs. Random Stroking Displays

The Raster Scan uses an electron beam to deflect a regular pattern across the tube. It is turned on and off, leaving a pattern of bright dots and blank spaces. These dot patterns form character representation on the screen. The dot matrix pattern may be made of 5 x 7, 7 x 9, 7 x 11, 8 x 15, or 11 x 13 character matrices. The total number of characters that can be displayed on the screen at one time has to do with the number of lines the monitor is capable of scanning across the face of the screen in a given period of time.

The random stroking display does not use the dot matrix concept. The deflection circuitry in a vector writing display moves the electron beam to the point at which the letter is to begin and creates the character on the screen via a series of short strokes or vectors. This technique was used to display a full page on the early word processing machines. Random stroking tubes do lend themselves readily to the display of line drawing and artwork as well as text. Therefore, they are most widely used in the display of computer graphic designs.

## Full Page Vs. Partial Page

The full page device costs more to build and is not an off-the-shelf TV monitor. This device provides for at least 60 lines of text on the screen and requires a faster cycle time to scan more lines across the display. Generally the applications that require the document to be viewed in its entirety can easily justify the costs.

The partial page device generally uses an off-the-shelf TV monitor and provides twenty four lines with a twenty fifth line reserved to communicate messages to the operator.

## Paging Vs. Scrolling

This feature in a word processor is very important and many vendors overlook this and therefore cause problems for the user, especially at revision time. Paging is moving a screen of information in its entirety either forward or backward. Some vendors only provide the capability of moving forward and to go backward the operator would have to start from the beginning. If paging exists then test to see how information, such as paragraphs, can be inserted and how the operating system handles repagination.

Scrolling is the most desirable because it provides the operator a means to view typewritten material in a continuous form. Most vendors today provide forward and backward scrolling which makes for easy deletion and insertion of material.

## The Keyboard

The keyboard is very important and should have the standard typewriter keys with upper and lower case letters. Look for the positioning of control and function keys and above all beware of the device that requires the operator to memorize numerous combinations to control the device. Most vendors now provide a separate key pad for numbers which may correspond to the card keypunch layout or the standard calculator board. The best test for the position style and touch of the keyboard is to field test by having the future word processing operator type text.

## File Management

This may be overlooked because of cost but evaluate the procedures that will be needed to control information once captured. Cassette or magnetic tape is inexpensive when compared to disk but the price an organization pays to log and retrieve information may exceed the original investment.

Disk storage, either floppy or rigid, forces the word processing operating system to log recorded information. Conversely, it also provides an easy access to the information, thereby decreasing operator effort and time. Many vendors provide for two disk drives, with one containing the operating system software and the other the data files. There are some vendors that house the entire operating system in the firmware of the processor and leave all disk storage for the user.

## Output

This area seems to be where new technology provides another variance with each tick of the clock. The old typebars are gone and the removable selectric "golfball" was a vast improvement. The industry adopted the interchangeable "daisy wheel" and the Qume and Diablos became the reigning kings. Each new idea fosters another. The character printers had versatility by changing the "wheel" or "ball" but this required a pause in the operation and a physical change of a component part. On the scene came the dot matrix printer which correlates to the raster scan concept except the dots are produced by wires or in the case of IBM the ink jet dot matrix. This new innovation provides great flexibility to control the size, shape and make up of characters through software. The future use of these devices is strictly up to the imagination of the operator and software capabilities.

The printers being designed today are departing from the ten or twelve to sixteen characters per inch. This new capability allows the user to stay with standard 8½ by 11 inch paper and eliminate the need for larger size computer printout stock. Proportional spacing is a feature that should not be overlooked. This allows a standard distance between all characters no matter what the shape of the character. The printers are rated by the number of characters they can print per second. The new dot matrix printers can attain a speed of 160 CPS while the Qume or Diablo range in speeds from 35 to 75 CPS.

## Functions and Features

Initial input to the word processor should be easy and fast. A single character delete key is mandatory to allow the operator to correct typo's on the fly. The device should allow for word wrap around so the operator does not have to worry about the right margin. There should be some visual indicators indentifying the left and right margins of the text. Additional features to look for are automatic centering and the ability to tab across the screen which can ease the load of the word processing operator.

The editing function is the heart of a good word processing machine and the techniques used by the designer determine the quality of the device. Editing can be performed easier on the CRT than with the LED or typewriter word processor. The cursor on the screen indicates where the action is to be performed. In the case of insertion, everything from the cursor to the end of text will be moved back to allow space for additional text.

In cases where text is to be deleted, the cursor is placed at the right location and by hitting a character, word, line, sentence or paragraph key the action is promptly performed. Another feature is the ability to position the cursor at the beginning of a block of text, thus identifying the start, then moving the cursor to the end of the block, noted by pressing the block key. The word processor will now isolate the block and be able to store or move the text to another location or file. Search and replace should be a standard feature and ideally there are different kinds of search capabilities in any system. A selective search is a simple short cut to get to a point in the file at which the operator wants to begin editing. The global search and replace performs the same function except that instead of stopping at the first occurrence, the device will replace data and continue until the end of the file is reached.

Justification, hyphenation and pagination are the greatest virtues of the word processing machines. A good system will provide the means to left and right justify text and display it as it will appear in typewritten form. This feature plus proportional spacing will will make any document neat and easy to read. Hyphenation is not standard between vendors and some systems offer automatic hyphenation while others will use a combination of hyphenation logic rules and a stored dictionary of limited size. Virtually all vendors offer some form of automatic pagination with the capability of generating running headers, footers and page numbers automatically. It is still up to the user to check the asthetic's of the systems page make-up decisions and to alter them if necessary.

There is an area that may be needed in some application areas which requires the use of sub-and superscript information. Ideally, this should be input at the terminal using one or two keystroke commands. Once entered, these features should be easily distinguishable to the operator.

## Conclusion

Word processors have excelled in the handling of repetative letter writing and document assembly tasks. Systems today are now offering relatively sophisticated selection, sort and merge functions that provide the user a great deal of flexibility.

Installations must define their need, then search for the device that best fits that need. The more you pay does not necessarily insure that all features will be there and careful analysis and evaluation will produce untold benefits in word processing. When evaluating word processors, vist the vendors and allow the person ultimately using the machine a chance to try it out.

# MONITOR: A Computer Based Management Information System for Special Education

Richard A. Whitney
Alan M. Hofmeister
Utah State University
Logan, Utah

## Abstract

MONITOR is a microcomputer based, management information system, which is designed to provide special education administrators with information concerning a school's compliance with applicable special education legislation. The system was originally conceptualized as a way to free special education staff from the heavy recording and documentation requirements of the Education For All Handicapped Children Act. No previous knowledge of computers or computer programming is required of users. All data entry and report generation activities are designed to be self-explanatory.

A central theme of the Education for All Handicapped Children Act, more commonly referred to as P.L. 94-142, is the guarantee that handicapped children will receive educational services that are designed to meet their individual needs. Once a child has been identified as being in need of special education services, the law provides for: (1) a nondiscriminatory assessment of the child's educational needs, (2) the development of an Individual Education Program (IEP) describing the services that will be provided to the child, and (3) a periodic review of the IEP to determine its appropriateness. These services must be provided within a general time frame established by law. Moreover, a child's parents or guardian must be notified and given the opportunity to participate in all aspects of educational planning (Federal Register, August 23, 1977).

Schools are required to provide documentation of the implementation of the mandated services and associated procedural safeguards. The benefits associated with the increase in services to handicapped children may be offset to some degree by the considerable increase in recordkeeping required of the schools (Morgan, in press). Schools must be able to document that the above services have been delivered in accordance with the established general timelines provided for in the law. Additionally, schools are required to maintain records of the number of children served and the nature of their handicaps. This information is used by schools to provide a rationale for funding requests and to be able to respond to program audits required of special education programs.

Administrators, teachers, and support staff are now required to spend considerable time in planning and documenting services to be delivered to handicapped children. Preoccupation with these recordkeeping tasks has reduced the amount of time special educators have available to implement direct teaching activities.

The Intermountain Inter-Tribal School (IIS), a U.S. Bureau of Indian Affairs boarding school, operates under legislation comparable to P.L. 94-142 (Federal Register, September 29, 1980). Given a relatively small special education staff and a relatively high percentage of referrals for special education (approaching 30% of the 600 student enrollment), special education staff members found it particularly difficult to provide services in accordance with the law. Due to the complexity of the compliance process associated with the delivery of services, there was concern that children might "fall through the cracks" in the school's monitoring process, with resulting service delays to students in need of special education services.

School administrators acknowledged the need for a systematic way of monitoring the compliance process for individual children. A monitoring system (which eventually became known as MONITOR) was conceptualized that would enable special education personnel to monitor and document compliance activities with a minimum of effort. A computer-based system was proposed that would reduce staff effort in recordkeeping and provide program managers with concise up-to-date information concerning the school's ability to maintain compliance with federal regulations.

## System Definition

The needs outlined by IIS personnel required the documentation of all major compliance activities for individual handicapped students as well as the provision of basic student accounting and diagnostic information to school administrators. IIS personnel determined that the system needed to be capable of performing the following tasks:

1. Assign and monitor timelines for all educational services to be delivered to individual handicapped students;
2. Provide timeline summaries of educational services delivered and those yet to be provided to individual handicapped students;
3. Provide program-wide information on the school's ability to deliver educational services within established timelines; and
4. Make basic student accounting information, such as student counts, available to program administrators.

The MONITOR system was designed, to meet these needs. The system adopted microcomputer technology to eliminate the need for extensive manual records. Microcomputer hardware was selected because of the desirability of a small, robust, self-contained system that could be used in other similar schools without much program modification. Consideration was given to the cost of other hardware alternatives such as timesharing. Timesharing costs would have exceeded the cost of the selected hardware over a relatively short period of time.

Development of the MONITOR system was accomplished by a project team from Utah State University. Operating on the IIS guidelines for the system, the project team selected a Northstar microcomputer system configured as follows: (1) Horizon Northstar DOS 5.0, (2) 64K of RAM, (3) 2 double density 5 1/4-inch floppy disk drives, (4) a SOROC IQ 120 video terminal, and (5) a Texas Instruments 810 RO bidirectional, dot matrix printer. This hardware configuration represented a fairly low-cost (approximately $5,900), powerful computing system capable of handling the proposed system and other educational applications, such as testing programs and other student accounting or recordkeeping systems that the school might wish to implement in the future. An additional feature of this microcomputer system was its high reliability. In previous use at Utah State University, hardware systems configured similar to the proposed system had required no servicing after extended periods of time.

## Software Development

Project management. Because of the developing nature of the compliance process, the development of the MONITOR system software had to be approached carefully by the project team. Figure 1 describes the major software developmental tasks. Each of these tasks was further subdivided and responsibilities assigned to individual team members.

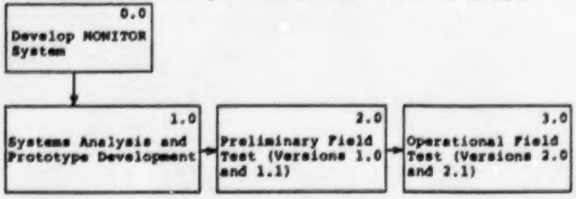
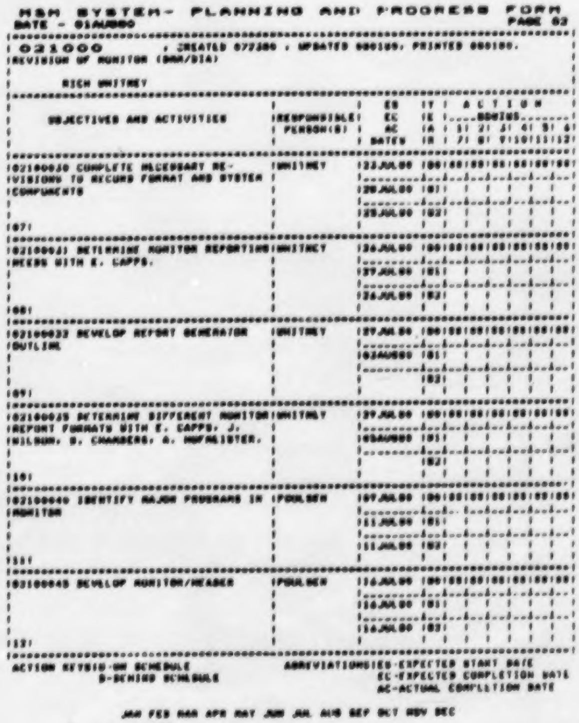

0.0
Develop MONITOR System
1.0
Systems Analysis and Prototype Development
2.0
Preliminary Field Test (Versions 1.0 and 1.1)
3.0
Operational Field Test (Versions 2.0 and 2.1)

Figure 1. MONITOR Developmental Sequence

Management of project accomplishments occurred through the use of a computerized management and monitoring system (M&M) in use at Utah State University's Exceptional Child Center. This system provided the project team with a listing of identified tasks and activities and facilitated the assignment of responsibilities and timelines for each task or activity. The kinds of information provided by the Management and Monitoring System are shown in Figure 2.



```
M&M SYSTEM- PLANNING AND PROGRESS FORM
DATE - 01AUG80                                             PAGE 02
 021000      , CREATED 072280 , UPDATED 080180, PRINTED 080180.
REVISION OF MONITOR (DHR/DIA)

    RICH WHITNEY

    OBJECTIVES AND ACTIVITIES      RESPONSIBLE    ES     T   A C T I O N
                                    PERSON(S)     EC     E  ....MONTHS......
                                                  AC     A   1  2  3  4  5  6
                                                 DATES   R   7  8  9 10 11 12

02100030 COMPLETE NECESSARY RE-   WHITNEY      23JUL80  00
VISIONS TO RECORD FORMAT AND SYSTEM            28JUL80  01
COMPONENTS                                     25JUL80  02
07

02100031 DETERMINE MONITOR REPORTING WHITNEY    26JUL80  00
NEEDS WITH E. CAPPS.                           27JUL80  01
                                               26JUL80  02
08

02100033 DEVELOP REPORT GENERATOR  WHITNEY     27JUL80  00
OUTLINE                                        02AUG80  01
                                                        02
09

02100035 DETERMINE DIFFERENT MONITOR WHITNEY    29JUL80  00
REPORT FORMATS WITH E. CAPPS, J.               05AUG80  01
WILSON, B. CHAMBERS, A. HOFMEISTER.                     02
10

02100040 IDENTIFY MAJOR PROGRAMS IN  POULSEN    07JUL80  00
MONITOR                                        11JUL80  01
                                               11JUL80  02
11

02100045 DEVELOP MONITOR/HEADER    POULSEN     16JUL80  00
                                               16JUL80  01
                                               16JUL80  02
12

ACTION KEYS:O-ON SCHEDULE            ABBREVIATIONS:ES-EXPECTED START DATE
            B-BEHIND SCHEDULE                      EC-EXPECTED COMPLETION DATE
                                                   AC-ACTUAL COMPLETION DATE

        JAN FEB MAR APR MAY JUN JUL AUG SEP OCT NOV DEC
```

Figure 2. Monitoring and Management System

Systems analysis. Development of the system was intended to follow a "top-down" developmental approach which incorporated considerable interaction between the project team and IIS personnel, particularly during the planning stages. Each component of the proposed system was checked out with IIS personnel to assure its match with the school's needs and federal requirements.

In analyzing the content for the system, the project team reviewed both the Bureau of Indian Affairs' Comprehensive Federal Regulations for Special Education and P.L. 94-142. The following major content areas were identified for the MONITOR system:

1. Activities related to the referral of students for special education services;
2. Activities related to the nondiscriminatory assessment of each student's educational needs;
3. Activities related to the development of an Individualized Education Program (IEP); and
4. Activities related to IEP annual review and re-evaluation.

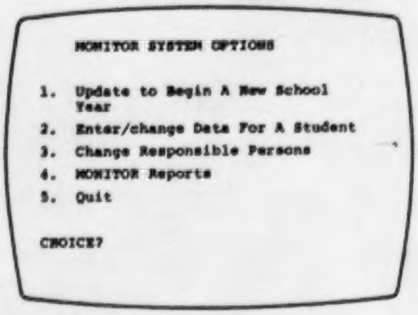
In each of these areas, school personnel were identified who would have the major responsibility for seeing that activities were completed. These persons included the special education coordinator, the assessment team leader, and the Child Study Team (CST) chairperson. The system was designed to track each handicapped student's progress through the network of compliance activities and indicate the extent to which related timelines for the completion of activities were being met by the personnel responsible for each activity.

Once the content of the proposed system was identified, the operational system components were identified by the project team. The system used the individual student as the basic record unit. In operating the system, users would create and access student records as well as prepare reports. Three basic program components were identified: (1) data entry and management, (2) report generation, and (3) system utilities. The project team elected to design the system around a "menu" format. Upon system start-up, users would be presented with a "menu." Each of the program components would be reflected in this menu. If the data entry system option was selected, the system would respond by presenting the user with a secondary "menu" of data entry options. The user could then elect to create a new student record or add information to an existing student record. Automatic error checking would be built into the system to prevent invalid data entry. If the report generation option was selected, the user would be presented a secondary "menu" of report options. Users could elect to produce reports for individual students or for responsible persons. If system utilities was selected, the user would be able to alter existing compliance statements (i.e., the way they would appear on reports) or timelines.

Upon completion of the description of system components, the project team held a series of meetings with IIS personnel to discuss the proposed system. IIS staff emphasized the need for users to be able to "interact" with the system without having to wait for reports or submit additions or changes to student records in a batch format. The project team indicated that the proposed system would indeed have these desired characteristics. Approval was granted by IIS personnel to begin the development of the prototype system.

Using the previously developed system diagrams and specifications, flowcharts were prepared for each of the operational components of the system. An internal critique of flowcharting was made by team members and coding begun. BASIC was chosen as the programming language to be used in the development of MONITOR. Each of the system components were brought into operation separately. Once a component was operating satisfactorily, another component was begun. Subsequent components were tested both separately and in conjunction with other components. This process prevailed throughout the development of the prototype.

<u>Description of prototype system</u>. The MONITOR prototype incorporated all of the features previously described. Following system start-up, users would be presented a "menu" with the system options listed in Figure 3. In addition to previously-mentioned options, two further system options were provided. The project team felt that the school should be able to "purge" the system's student records of all students who either were no longer receiving services or who had left school at the end of the year. This capability was referred to as updating to begin a new year. The ability to change the names of responsible persons serving as assessment team leaders or CST chairpersons was also seen as a necessary system option.

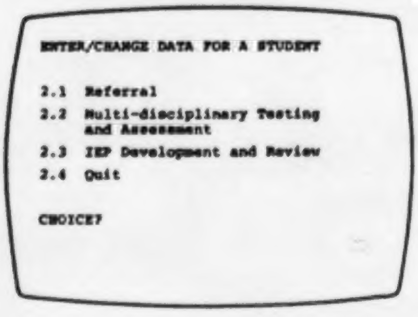


```
     MONITOR SYSTEM OPTIONS

1.   Update to Begin A New School
     Year
2.   Enter/change Data For A Student
3.   Change Responsible Persons
4.   MONITOR Reports
5.   Quit

CHOICE?
```

Figure 3. MONITOR System Options

Selection of "1. Update to Begin A New School Year" caused the system to respond by asking for the current date. Once this had been accomplished, the system would scan all records in the system data files and select those students who were no longer receiving special education services. The system would then produce a report on the printer listing these students and the date services were terminated. When the report had been completed, the "menu" of system options was again displayed on the screen. The user could then select another system option.

The system option, "2. Enter/change Data For A Student" caused the system to display a secondary "menu" with the information shown in Figure 4. The user could then


ENTER/CHANGE DATA FOR A STUDENT

2.1 Referral
2.2 Multi-disciplinary Testing and Assessment
2.3 IEP Development and Review
2.4 Quit

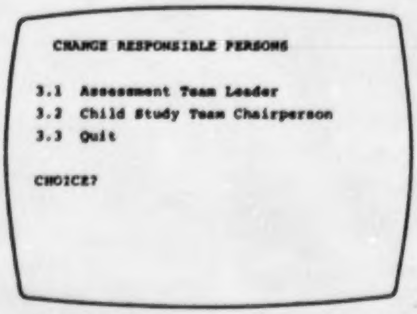
CHOICE?

Figure 4. Enter/change Data For A Student Options

create, add, or change information within any of the areas shown. If the user wished to create a new student record, "2.1 Referral" was selected and a "menu" listing of all referral descriptive information and compliance activities was displayed. The user then selected the information or compliance activity required, and a prompt appeared (i.e., Student Number . . .[ ]). The user entered the correct information and was then returned to the list of referral information and compliance activity options. Automatic error checking was incorporated in all phases of data entry. If invalid characters were entered, an error message appeared momentarily and the user was requested to re-enter the information.

If the user selected "2. Multi-disciplinary Testing and Assessment" or "3. IEP Development and Review," different listings of compliance activities were displayed. The user would interact with the system as described above. In all, the MONITOR prototype allowed users to enter data for 31 compliance activities associated with referral, multi-disciplinary testing and assessment and IEP development and review.

If a user selected the system option, "3. Change Responsible Persons," a secondary "menu" was displayed, with the information shown in Figure 5. Users could change names of responsible persons for individual students using this option.

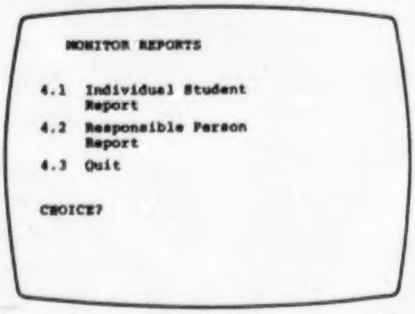

CHANGE RESPONSIBLE PERSONS

3.1 Assessment Team Leader
3.2 Child Study Team Chairperson
3.3 Quit

CHOICE?

Figure 5. Change Responsible Persons Option

Selection of system option, "4. MONITOR Reports," again caused the system to display a secondary menu. Figure 6 shows the report options available to a user. Two basic report options were available. The individual student report produced a listing of all 31 compliance activities for a particular student and indicated for each whether it was completed, in progress or behind schedule, and supplied the completion date, if appropriate. This report was designed to be used to document the school's activities on behalf of the student in question. Additionally, it could be used by the special education coordinator, the assessment team leader, or the CST chairperson to assist in program planning for the student.


MONITOR REPORTS

4.1 Individual Student Report
4.2 Responsible Person Report
4.3 Quit

CHOICE?

Figure 6. MONITOR Reports Option

A second report was available that listed a responsible person and the compliance activities for which the person was responsible. For each compliance activity, the total numbers of students completed, in progress and behind schedule, were displayed. This report was designed to provide diagnostic information to responsible persons and program managers as to probable "trouble-spots" in the compliance process.

The final system option was "5. Quit," and allowed users to sign-off the MONITOR system. Selecting this option caused the screen to clear. Users would then turn off the terminal, microcomputer and printer.

*Prelimininary field-test*. Upon completion of the prototype MONITOR system, IIS personnel were contacted by the project team. A location for the system hardware was selected and times arranged for in-service training of IIS personnel in the use of the system.

The system hardware was located in the special education coordinator's office. The coordinator selected one secretary to serve as the principal operator of the system. Project team members installed the hardware and conducted limited in-service training with the coordinator and the principal user. From the outset of the field-test, IIS personnel were instructed to refer all questions and problems directly to the project team at Utah State University via a toll-free WATS line.

The project team at Utah State University maintained a duplicate system (both hardware and software) to allow for double-checking questions and problems encountered by users. During telephone conversations or immediately following a telephone contact with IIS personnel, project team members attempted to verify problems and questions that users had and make appropriate changes, if necessary.

The duration of the preliminary field-test was set for three months. Field-testing began in March, 1980, and concluded in May, 1980. No attempt was made to place all handicapped students on the system. Project team members suggested that 20-30 students be placed on the system initially. The criteria for selection of the students to be placed on the system were: (1) the student had just been referred to special education services, and (2) coordinator judgment that the students would proceed fairly rapidly through the compliance process.

Data collection during the preliminary field-test involved the identification and categorization of system problems and observing the principal user enter data and produce reports. The project team also observed the entire compliance process as it was organized within the coordinator's office. The forms used by the special education program were reviewed to determine their match with the prototype system. Shortly after initiation of the field-test, it was found that certain significant differences existed between the actual process being adopted at the school and the way that the MONITOR system handled the compliance activities. Project team members met with IIS personnel to resolve these differences. As a result, certain revisions were made to MONITOR (Version 1.0) with respect to the wording and sequence of the compliance activities as they appeared on the system. The resulting prototype, MONITOR (Version 1.1) was used throughout the remainder of the preliminary field-test.

At the conclusion of the preliminary field-test, project team members analyzed the data collected. Based upon an analysis of the data records placed on the system while at IIS, and feedback from the principal user and others, the following problems were identified:

1. The prototype system, as organized, allowed users to skip required compliance activities, enabling them to circumvent the compliance process;
2. Student records contained a number of data entry errors. These data entry errors had been accepted by the system and resulted in report errors;
3. There were inconsistencies between data input documents currently used by the school and the specific input requirements of the system;
4. There was a need to incorporate ways to delete student records and to make provisions for students on "inactive status;"
5. IIS personnel indicated that the current reports provided by the system were inadequate. The information appeared to be available, but the report options were not able to access it;
6. Compliance statements and associated timelines were inadequate and provisions for user modification of timelines needed to be added.

The problems identified above strongly suggested that major changes be made in MONITOR (Version 1.1). Project team members initiated a system update in June, 1980. The concept of the "menu" approach was preserved in the updated version of MONITOR, but control over the access to compliance activity data entry was turned over to the system. Once a student was referred for special education services, a general sequence of compliance activities was identified. This sequence began with referral activities and proceeded through IEP development and into review and re-evaluation of student programs. The system was modified to automatically prompt users to initiate annual review and re-evaluation processes. Users would be required to document the completion of an antecedent compliance activity before a successor activity could be recorded. Users could no longer circumvent the compliance process. This appeared to one of the most significant contributions of the revised MONITOR (Version 2.0).

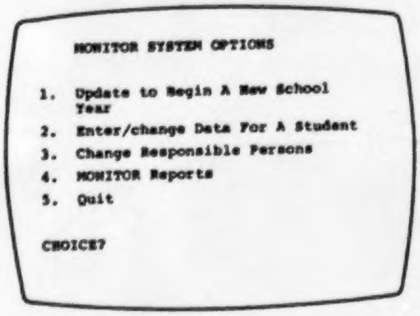
MONITOR (Version 2.0) also incorporated more sophisticated error checking procedures. Users were also given the capabilities altering the content of the compliance activity statements. Other features, such as user modification of timelines and the ability to delete or place students on "inactive status," were noted but not incorporated into MONITOR (2.0). The project team felt that such features would affect the data obtained during the operational field-test. These features would be incorporated in future versions of MONITOR but were not incorporated in the present version.

Operational field-test. MONITOR (Version 2.0) was re-installed at IIS in October, 1980, for the operational field-test. Project team members conducted in-service training with both the principal user (the coordinator's secretary) and the coordinator. For the operational field-test, thirty students were selected in a manner similar to that employed in the preliminary field-test.

The operational field-test involved data collection in the form of user observation, input document review, spot checking of individual student records and documentation of problems that arose. The operational field-test placed emphasis on the improvement of input documents, the development of user documentation based upon in-service training experience, and the correction of minor system errors. This process, along with the final evaluation of the impact of the MONITOR system will be concluded in May of 1981. All observations suggest that the project is meeting the objectives set for it.

## MONITOR (Version 2.1) Description

Overview. Since the implementation of Version 2.0 in the operational field-test, a number of minor changes have been made in the MONITOR system. The capabilities for deleting and placing a student on "inactive status" as well as the ability to modify compliance timelines have been added. Users now have the option of displaying a short description of the MONITOR system on the terminal screen. This feature has been added to assist persons unfamiliar with the system in understanding its capabilities. Reporting capabilities have also been expanded. These changes have been combined to form the current Version 2.1. Figure 7 shows the system options available to users of the current MONITOR system. It is this version that is described in detail below.



MONITOR SYSTEM OPTIONS

1. Update to Begin A New School Year
2. Enter/change Data For A Student
3. Change Responsible Persons
4. MONITOR Reports
5. Quit

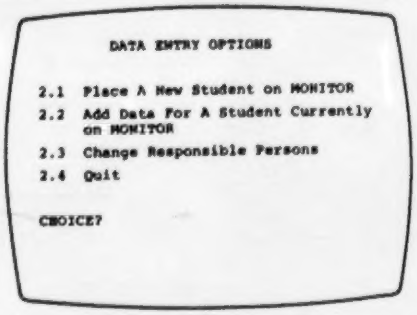
CHOICE?

Figure 7. MONITOR (Version 2.1) System Options

Data Entry. User data entry has been simplified and the sophistication of error checking has been increased in Version 2.1. System options as well as sub-options are selected from "menus" in the same manner as in previous versions. Automatic error checking occurs for all user input. Whenever an input error occurs, the system displays a brief error message for the user and returns to the point at which the error was committed. In general, two types of information are placed on the system by users. Descriptive information, such as name, student number, tribe, and referral source constitute one type of information placed on student records. The second type of information placed on student records is date information concerning the completion of compliance activities. Both types of information involve the use of standard screen formats. Descriptive information is requested by providing the user with prompts indicating what information is needed (i.e., Student Number . . . [ ]). Once the information is entered, the system presents the next information prompt. When all required descriptive information has been entered, the system displays the information and allows the user to edit and correct input. When the user is satisfied with the information shown, the system is directed to place the information on the student's data record.

The date entering screen format is similar to that used for entering descriptive information. Users are required to enter dates by first entering the month upon prompt (i.e., Month . . . [ ]), then year and day. Once the date is entered, the system checks to determine whether it is a valid date. If incorrect, the system displays an appropriate error message and returns the user to the location where information must be corrected. If the date has been judged to be valid, the system allows the user to make final alterations in the date and then places it on the student's data record. The descriptive information screen format and the date entering screen format are used together in series to enter all data for students on the system.

If a user selects the system option "2. Input Data for a Student," the system displays a secondary "menu" with the information shown in Figure 8. This secondary "menu" shows all of the data entering procedures available for students on the system. Users may elect to place a new student on the system, input data for a student currently on the system or change responsible persons for a student currently on the system.



DATA ENTRY OPTIONS

2.1 Place A New Student on MONITOR
2.2 Add Data For A Student Currently on MONITOR
2.3 Change Responsible Persons
2.4 Quit

CHOICE?

Figure 8. Data Entry Options

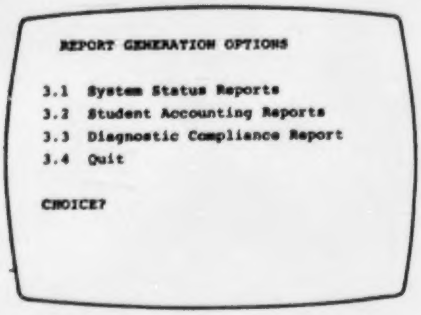
If a user selects the option, "2.1 Place A New Student on MONITOR," the system responds by displaying the descriptive information screen format. The user enters the student's name and student number first. The system checks the student number entered to determine whether the number has already been used for another student or whether this student has already been entered on the system. Assuming that the number has not already been entered, the system proceeds with prompt-student information.

Once the descriptive information has been entered on a student's data record, the system continues by asking questions relating to the completion/noncompletion or decision/no-decision concerning compliance activities. Questions answered in the affirmative by the user result in the display of the date entering screen format, which is provided to allow the user to enter the date on which a decision was made or the activity completed. A question answered in the negative indicating that a compliance activity has not been completed or a decision not made, results in the system displaying the message, "This Task Must Be Completed Before Data Entry May Continue." The system then returns the user to the system options "menu." Further data entry is not allowed until the compliance task is completed and a date entered. This prevents users from circumventing the compliance process by skipping certain compliance activities.

If a user selects the secondary option, "2.2 Add Data For A Student Currently on MONITOR," the system responds by requesting the student's name and student number. The student's data record is accessed and searched to locate the last compliance activity for which a completion date was entered. The system then displays the compliance activity question and requests the user to answer whether this activity has been completed. The sequence of activities from this point on is identical to that described above.

A user selecting the secondary option "2.3 Change Responsible Persons," is again requested to indicate which student and student number is desired. The system then prompts the user to indicate which responsible person is to be changed. The user enters the responsible persons's title (i.e., assessment team leader or CST chairperson) and the person's name. Once this is done, the system changes the appropriate entries in the student's data record.

Report generation. The reporting capabilities of Version 2.1 have been greatly expanded from previous versions. IIS personnel provided considerable feedback to the project team at the conclusion of the preliminary field-test. The result is a more concise, meaningful, and useful set of report options. A user selecting the system option, "3. Product MONITOR Report(s)," is presented with a secondary "menu" of general reporting options. These options are displayed in Figure 9. Three levels of system reports are available on MONITOR (Version 2.1). Within each level, one or more individual report options may be accessed.


REPORT GENERATION OPTIONS

3.1 System Status Reports
3.2 Student Accounting Reports
3.3 Diagnostic Compliance Report
3.4 Quit

CHOICE?

Figure 9. Report Generation Options

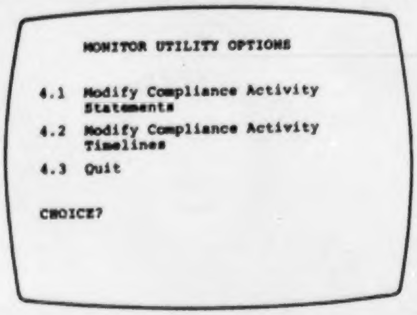
If a user selects the secondary reporting option, "3.1 System Status Reports," another "menu" of options is displayed by the system. Users may select from a Student Status Report, an Assessment Team Leader Status Report, or a CST Chairperson Status Report. The Student Status Report displays the compliance activities for a given student and indicates for each activity whether it has been completed or is behind or on-schedule. This report also provides the expected starting date, the expected completion date, and where applicable, the actual completion date. Expected dates are calculated based upon timelines built into the system.

The Assessment Team Leader Status Report and the CST Chairperson Status Report both display the compliance activities for which each responsible person is accountable. For each compliance activity, the percentage of students who are completed, behind schedule or on-schedule, are displayed. This information allows a responsible person to determine how well he/she is meeting compliance timelines for the provision of educational services.

A user selecting the secondary reporting option, "3.2 Student Accounting Reports," is able to select from five different BREAKDOWN reports. Cumulative reports displaying students by (1) age x handicapping condition, (2) age by assessment team leader, (3) age x CST chairperson, (4) age x tribe, and (5) tribe x handicapping condition are available to users. These particular reports have been requested by IIS personnel and provide them with information needed for reporting and funding agencies.

Selection of the secondary reporting option, "3.3 Diagnostic Compliance Report," allows the user to produce a summary report across all of the accessible compliance activities. For each compliance activity, the total number of students on and behind schedule and those who have completed the activity, are listed. This report is useful for the special education coordinator or other school administrators to assist in assessing trouble spots in the program's overall compliance.

System utilities. This is the final system option available to users. Users selecting "4. MONITOR Utilities" may select from the "menu" options displayed in Figure 10.


MONITOR UTILITY OPTIONS

4.1 Modify Compliance Activity Statements
4.2 Modify Compliance Activity Timelines
4.3 Quit

CHOICE?

Figure 10. MONITOR Utility Options

System utilities allows users to modify compliance activity statements included in the system reports or change the timelines upon which the system is based. These options were not included in previous versions of the MONITOR system. It was felt by project team members that IIS personnel given these capabilities may inadvertently change compliance timelines, resulting in the system monitoring timelines that significantly differed from the law. Since the major purpose of the system was to keep the school within compliance standards, these capabilities were withheld until Version 2.1.

The ability to alter compliance statements and timelines allows program personnel to "individualize" a compliance system. MONITOR keeps track of over fifty compliance activities in the current version. The law obviously does not provide definite timelines for each of these activities, but does give general timelines. Therefore, the school can modify timelines for some of the interim compliance activities without affecting the school's compliance with the law.

Once the system has been in operation for a period of time, the special education coordinator can access the "Diagnostic Compliance Report," which provides an overall view of the school's compliance for each of the compliance activities monitored by the system. Using this document as a planning tool, the coordinator can analyze which activities have positive slack and which have "negative" slack. Timelines may then be adjusted on interim compliance activities. This allows the special education program at the school to operate more smoothly while maintaining compliance with the law's timelines.

## Benefits and Future Directions

The benefits of a system, that monitors a complex series of activities in a program area where practices are still in development, are twofold. First, the usual benefits of using a computer's power for automated recordkeeping are available. The extent of this benefit is related to the quantity and complexity of the recordkeeping task. Currently, at the Intermountain Inter-Tribal School, special education staff are required to monitor some 45-50 specific due process activities for each of the 150 students who are receiving special education services. This yields some 6,700-7,500 individual activities that require monitoring and documentation by special education staff.

The second benefit of implementing a system such as MONITOR is that, in the process, school personnel are required to clarify tasks, personnel assignments and timelines. This clarification is often a difficult process. The interaction between the project team and IIE personnel resulted in the clarification of special education practices and associated compliance tasks, personnel assignments and timelines. The importance of this facilitation and clarification role of computer-based projects can be quite significant and is often overlooked.

The project team has identified two major areas for future development. Currently, project team members are working on the development of a MONITOR system that will operate on a smaller microcomputer system. The Apple II microcomputer system has been selected for use in this development. It represents a less expensive alternative to the Northstar system and has a significant potential for use in a myriad of other educational applications. such a system could be adopted by many small schools previously unable to afford the luxury of computer support. The translation of the MONITOR system to other hardware systems represents an attempt by the project team to make the system as exportable as possible within the educational community.

The second area of future development surrounds the application of microcomputer technology to the actual development of IEPs, development of classroom level computer-based curriculum management systems and associated uses in educational decision making. These areas of development, when integrated with the current MONITOR system, would allow educational personnel more time for their teaching responsibilities while increasing the quality of required recordkeeping duties.

## References

Federal Register, Washington, D. C.: U.S. Government Printing Office, August 23, 1977.

Federal Register, Washington, D. C.: U.S. Government Printing Office, September 29, 1980.

Morgan, D. P. A primer on individualized education programs for exceptional children: Strategies and practices. Reston, VA: Foundation for Exceptional Children, in press.

NOTE: The authors wish to express their gratitude to Mr. David Burch, Mr. Charles Geboe, Ms. Ethel Capps, and Ms. Sarah Chambers of the Intermountain Inter-Tribal School for their assistance in the development of MONITOR.

# MICROCOMPUTERS: INTEGRATION INTO THE CURRICULUM

Donald E. Wiederanders
Lynn C. Schwandt
Price Laboratory School
University of Northern Iowa
Cedar Falls, Iowa 50613

In 1975 the National Advisory Committee on Mathematics Education called for "the development of modules that integrate computers and regular mathematics courses." This is a description of a project designed to do just that. Existing software will be modified and new software will be created to complement the teaching/learning process for the contemporary mathematics curriculum (grades 9-11).

1 Microcomputers: Integration into the Curriculum

The Iowa Microcomputer Curriculum Project proposes to make the microcomputer accessible as an adjunct to the contemporary curriculum. Phase I of the project will focus on microcomputer programs for contemporary algebra I, geometry, and algebra II courses.

## THE PROBLEM

The microcomputer is rapidly becoming a standard piece of equipment in today's society. Though only approximately 500,000 were sold in the United States in 1978, it has been estimated that more than 1,200,000 were sold in 1980.[1]

The potential impact of the microcomputer on the lives of the students who are now in our schools is staggering. Their lives will be affected first by the improved quality of education available to them during their school years. Second, their post-school years will be significantly impacted by changing job opportunities, drastically altered job descriptions, and unpredictable modifications of the parameters that affect even the least technically oriented life styles. The invention of the digital computer has rapidly propelled the worlds of business and industry into the electronic age. The advent of the low cost, highly versatile microcomputer is expected to have the same effect on the farm, the small business, and the home.

As early as 1975, F. James Rutherford, director of the National Science Foundation, in the 2 foreword to the collection of papers, Technology in Science Education: The Next Ten Years,[2] writes "...it (the computer) is a versatile tool that has affected not only every field of science, but also has come rapidly into common household use. The computer permeates many aspects of our society and impacts significantly on the daily lives of most people."

The secondary schools apparently are aware that the microcomputer has tremendous potential for education. Many have already made some level of commitment to explore ways in which the microcomputer can be incorporated into the instructional process. In spite of the squeeze put on schools by inflation, declining enrollments and taxpayer concern regarding the rising cost of education, schools are increasing their commitment in the area of microcomputers. Thousands of attendance centers presently have one or more microcomputers. Others are in the planning stages for microcomputer usage.

The students in the schools that have access to microcomputers are rapidly becoming skilled users of existing programs and many are entering the growing ranks of enthusiastic programmers. The problem is to provide these children and others with appropriate programs that will capitalize on the microcomputer's capability to complement the learning process. "One of the most pressing problems for the classroom teacher whose school has purchased a microcomputer is integration. ...ways in which a microcomputer can be

---

[1] LaFrenze, Dale, talk presented at Association for Educational Data Systems, regional meeting October 23-24, 1980.

[2] Prepared for the National Science Foundation, 1979.

3 integrated into real world educational settings."[3]

Charles L. Blaschke, President of Education TURNKEY Systems and Director of Microcomputer Education Application Network, conducted an informal survey of "interested" users at six national education conferences in 1979. He reported, "The major bottleneck limiting the widespread and effective use of microcomputers in elementary and secondary schools in the availability of quality software (programs) which meet high priority user needs."[4] A monograph produced for the Conference Board of the Mathematical Sciences by the National Advisory Committee on Mathematical Education[5] includes the following statement, "The development of modules that integrate computing and regular mathematics courses is a sensible first step in realizing the extensive impact of technology on the methods of mathematics ..." This basic premise is reinforced by a statement approved by the Board of Directors of the National Council of Teachers of Mathematics, "Educational decision-makers including classroom teachers should seek to make computers readily available as an integral part of the educational program."[6]

The solution of the problem for the entire K-12 curriculum can become attainable only after an operational model is developed. The Iowa Microcomputer Curriculum Project - Phase I addresses itself to a manageable portion of the total problem. The long-range objective for this project is to develop a prototype that will be useful in addressing the problem at other levels of mathematics and in other subject areas of the K-12 curriculum.

The wide-spread acceptance of the microcomputer has created a rapidly growing market for programs that allow the user who has a very minimal amount of operational skill and no technical or programming expertise to interact directly with the microcomputer. Dozens of commercial sources are flooding the market with programs that vary greatly in quality and utility. Written descriptions of these programs unfortunately are seldom, if ever, available for review prior to purchase. With thousands of programs to choose from and no way to adequately identify quality programs, the school finds itself in a serious dilemma. This dilemma is compounded by limited funds for the purchase of microcomputer programs and an even more limited capability for matching appropriate programs with curricular needs.

## THE PROJECT

The Iowa Microcomputer Curriculum Project - Phase I proposes to establish a process which will: a) make it possible for the teachers of algebra I, 4 geometry, and algebra II to match quality microcomputer programs with specific content modules[7] within the existing curriculum; b) inform potential users of the availability of curriculum oriented programs for the microcomputer; c) provide guidance for authors and potential authors by identifying content areas for which quality programs do not currently exist.

The project proposes to accomplish this by developing a matrix. Each cell of the matrix will identify a specific content module and a developmental level[8] within the learning of that content. One dimension of the matrix will consist of a composite list of the mathematics modules that are common to a large majority of the algebra I, geometry, and algebra I curriculums. The second dimension of the matrix will be the three stages of concept development considered most likely to utilize microcomputer programs as an adjunct to the instruction/learning process. These stages will be identified as: (1) introduction, (2) skill development, and (3) application. Thus, each mathematical topic appearing in the matrix in vertical sequence will have three cells to the right of it. Within each of these cells, several types of programs and the sources for those programs may be listed. An empty cell would alert prospective authors to the fact that this content area needs a program. As presently planned, the matrix would be an extension of Figure 1. As the project develops, many of these cells will have listed within them quality microcomputer programs and sources for those programs. It is this matrix that will be sent to classroom teachers for use as a guide in acquiring and using microcomputer programs as part of the instructional process.

| Concept Development Stage → <br> Content Module ↓ | Introduction | Skill Development | Application |
|---|---|---|---|
| | | | |
| | | | |
| Addition and subtraction of integers | AI-42 Vector Model (Source) | | AI-72 Using a guessed mean (source) |
| | | | |

Figure 1

A critical part of this project relates to the process used to determine whether a submitted program shall be listed in the matrix. The quality of the programs in the matrix must be assured by a carefully designed and consistently executed evaluation process. Submitted programs will be screened by the project manager, reviewed by at least two members of an editorial panel and finally critically evaluated by at least two referees. This process will be designed to insure that each program publicized by the matrix is not only technically and pedagogically sound but recommended by competent educators for use in teaching a specific content module.

---

[3] Keyser, E.L., "The Integration of Microcomputers into the Classroom or Now That I've Got It, What Do I Do With It?", AEDS Journal, Vol. 13. Fall 1979, p. 114.

[4] Blaschke, Charles, "Microcomputer Software Development for Schools: What, Who, How?", Educational Technology, October 1979, Vol. X, IX, No. 10, p. 26.

[5] National Advisory Committee on Mathematics Education, Overview and Analysis of School Mathematics - Grades K-12, pp. 39, 40, 1975.

[6] "Computers in the Classroom", The Mathematics Teacher, March 1978, p. 468.

[7] A three- to-ten-day content focus that relates to a specific mathematical skill or concept.

[8] Adapted from the instructional steps attributed to Herbart in Examples of Teaching Method by Brodie and Palmer, Rand McNally, 1965, p. 130.

The development of the program evaluation process will involve careful study of the processes used by other groups that are struggling with the task of evaluating microcomputer programs. The check lists and criteria used at each stage of the evaluation process will be critiqued by external reviewers prior to use. The evaluative instruments themselves will be subject to modification as a result of feedback from both program referees and classroom teachers that use programs from the matrix.

Quality begets quality. There is an obvious need to have the first matrix contain quality programs that will contribute directly to the instructional process. The quality identified with the programs found in the first matrix distributed will establish expectations for users and set standards for future authors. A review of currently available programs is not encouraging. Many of the existing programs were apparently designed to intensify the user's interest and/or demonstrate the capabilities of the microcomputer. Though some existing programs address themselves to content, few, if any, can be expected to match the specific instructional needs as identified within the matrix. Since having quality programs for inclusion on the first matrix is crucial to the overall success of the project, steps must be taken to insure their existence and availability. To that end the Iowa Microcomputer Curriculum Project - Phase I will conduct two authors (programmers) workshops during the summer of 1981.* The major objective of these workshops will be to produce microcomputer programs that fit the matrix.

An effective process for recruiting authors and users must be an integral and ongoing part of the project. An aggressive campaign to recruit authors for the authors workshop will be conducted within Iowa and adjacent states. Following the workshops, quarterly mailings will be made which include the latest matrix update, suggestions regarding its use, and promotional material regarding authorship. This distribution will include all secondary schools in the state of Iowa as well as target groups and interested people throughout the nation. Two program users workshops will also be held, one in the fall and one in the spring.

---

*see attached

# THE COMPUTER IN DEVELOPMENTAL EDUCATION

Jane H. Wilde
Director of Curriculum Development
Voorhees College
Denmark, South Carolina 29042

This paper focuses on issues relating to the use of the computer with developmental students that should interest developmental educators and administrators. Opinion is divided on the efficiency of computer use as an instructional tool for developmental students. Faculty resistance and cost factors are two important initial considerations. Other significant issues regarding computer instruction for developmental students relate to the need for a competent student to have some degree of computer literacy because of the increasingly wide-spread use of computers. The paper explores the meaning of computer literacy in today's society. Many educators allege that developmental students with their deficiencies in basic skills are blocked by computer anxiety from attaining a requisite literacy. We contend that minimal exposure to even simple computer assisted instructional programs will rob the computer of its mystery, provide training in basic skills, and provide the computer sophistication required for survival in tomorrow's world.

One of the nagging problems of higher education is the large number of developmental students found on campuses across the nation - students with native ability, who for one reason or another have not reached the levels of basic skills competence traditionally expected of the college student. Another factor that is increasingly a fact of modern life is the explosion of computer use across American life. When these two facets are viewed side by side, there is cause for deep concern about future developments in American education, particularly since education has tended to be a conservative institution in society. This is not to suggest that there has been no concern for developmental students or computers, but that the use of computers by developmental students (and probably many students) in instructional programs has certainly not reached its full potential in higher education. This lag has profound social and political ramifications, since these students are very often members of minority groups in society who are least able to afford access to computers. The problem is compounded since it is apparent that we may at last be on the verge of a drastic and revolutionary change in the delivery of education because of the computer. The coming changes have been described as similar in impact to the changes in the institutional paradigm produced by the printing press. The emergence of the microchip has the potential to produce a decentralization of education and to cast faculty into a radically different role.

The major focus of this paper is to discuss briefly the role of the computer in developmental education in higher education, some issues relating to it, and to provide further grounds for reflection on the issue of computer literacy as a significant national problem. It is approached from the point of view of an administrator/faculty member working with instructional development. It seeks to focus on a range of issues that should be considered relating to the use of the computer by developmental students and of how far developmental students can go in attaining a degree of computer literacy. Developmental students in college come with low levels of achievement in basic skills areas (reading, computation, writing) and with low self esteem. These factors would seem to militate against such students progressing in the development of the ability to work with computers, feel comfortable with them, and to appreciate the significance of their role in our rapidly changing world.

However, evidence suggests that the use of computers with basic skills students can be successful and that they can move on to the ability to utilize the computer as a tool for improving their status. A number of programs in a variety of settings provides evidence for this. The programs with basic skills students at Xavier University of New Orleans, the uses made at Bennett College in North Carolina, to cite but two institutions, are successful. The PLATO programs in basic skills have been successful in raising the skill level of developmental students. Other programs undertaken with semi-literate inmate populations in prisons demonstrate educational achievement. The summary of the use of CAI with short term prisioners in the Bexar County

Detention Center in San Antonio is instructive.

Perhaps we can take as a given the fact that the use of CAI is effective and advantages for developmental students. Although an ETS study has shown that students tend to achieve at much the same level with or without computer assisted instruction, the computer does provide significant advantages when used in basic skills programs. It provides the instructor relief from the drudgery of working through drill and practice sessions with the student. The computer can tirelessly act as the patient drill master. It can provide immediate feedback to the student. A well developed program will pinpoint the cuase of the error and direct the student to the specific remediation he needs. In addition, given the low self concept of most developmental students, the computer does not publicly embarrass the student when he makes an error and reveals the extent of his deficiencies. Further, the positive feedback provided to the student on his successes increases his self concept and self confidence. The computer can keep records of the student's progress for the instructor. This information can assist the instructor in providing more individualized attention for the student as the instructor is freed from the drill and practice routines. As the student becomes involved in the interactive dialog with the program (or perhaps, better, with the persons who developed the program), the student can become motivated to try to beat the computer, to try to get the positive comments provided for successful performance.

However, there are other issues worth raising related to the role of the computer in forcing the improvement of developmental programs. At present, there are really no programs that provide for the certification of faculty trained to teach the basic skills of developmental students much less CAI. There is much disagreement about what approach works best with these students. As one ponders the problem, a number of causative factors seem involved - the advent of TV as a universal medium of communication, relaxation of standards (some would argue). A look at the psychological factors involved may be more instructive. In most instances, a low level of self esteem and self confidence seems to be a problem for developmental students along with the current malaise of the modern world - a sense of alienation, of not belonging. This may result in the student either holding an exaggerated view of achievement or giving up too easily because he feels it is beyond his capabilities. Many educators such as Meeth and Rouche have pointed to the importance of the building of groups, the awareness of a shared problem, the importance of group work, and a firm but sympathetic discipline in improving achievement levels. These dimensions are often overlooked in working with basic skills students who are often thought of as lazy and incapable of being motivated - at least by traditional means.

Questions of the cognitive styles of developmental students are also involved. Is CAI equally effective for all students and in line with their cognitive styles? Evidence from a short-term experience with a small group of students at Voorhees College under a Lilly Endowment grant for improvement of instruction in the social sciences suggests that there may be a preference for both group oriented and more intuitive learning among many developmental students. This would seem to make the more structured logical approach of most present CAI programs less well suited to these students. Certainly the issue of cognitive style deserves further consideration as educators consider CAI.

Indeed, experience with the development of CAI can lead to further improvement in the instruction of developmental students. Although there may be insufficient evidence at present to support the contention, it may be that the problems of the so-called developmental students may be the result of deep-seated emotional problems related to lack of success in academic learning. the work done by Shelia Tobias and other on math anxiety tend to suggest that there may well be a learning anxiety. Therefore attention to the structure and discipline of the computer assisted instructional program along with a focus on the emotional aspects of the problem could lead to significant improvements in developmental education. Obviously, this calls for an alliance of educators in content areas who are also deeply interested in how people learn.

A computer assisted instruction program can incorporate elements of group learning and adapt itself to the more visual world that has come with TV. As computer assisted programs develop, and as educators are forced to rethink the structure of their discipline, different paths toward competence in the discipline can be identified. It has been observed that the structures associated with many disciplines are in part a result of the way in which they have been taught. Stephen C. Lower of the University of British Columbia has pointed to the need for more flexibility. "Because CAI can introduce so many new dimensions to learning, it is important that we take a careful look at our own subjects from these different viewpoints. Ultimately one would hope that it will become possible to emphasize different kinds of structual development that will correspond more closely to the needs of individual students. As one moves to more sophisticated terminal devices that are able to produce graphics and film displays, animated motion and video responses, the possible ways of structuring the subject increase enormously, and present the possibility of developing alternative course structures that attend to different cognitive styles." However, one of the drawbacks of many CAI basic skills programs that have been developed is that they lack the desired degree of interactive programming and tend to emphasize the lower level cognitive skills. Caldwell has pointed out that many programs tend to be merely drill and practice and perpetuate many of the worst features of today's classroom practice. At present, indications are that very few scholars have the ability to translate the higher level academic concepts into software utilizing both computer and/or telecommunication capacity.

It is apparent that programs for developmental students could move from graphics designed to produce understanding into the more traditional word-oriented approach. If a picture is worth a thousand words, it is well worth its use as a motivator of students. If referrant teaching works well with developmental students, then programs using graphics, role playing, and simulation can be developed to provide the pathway by which the student can move from things in his experience to an understanding of a concept. It is better to show the student than to tell him.

The foregoing comments point to a number of problems related to CAI and developmental students. Further, it has focused on CAI as a tool for helping to achieve traditional literacy. However, this is no longer sufficient, given the increasingly widespread use of computers in society. It appears that a requisite for successful living in the near future will be computer literacy. It is not clear what public attitudes presently are relating to computer literacy. (No such survey has been taken since 1971.) But the growing debate over what constitutes computer literacy, is an indication of increased concern among those best acquainted with the computer as an educational tool. The recent conference in Reston on National Computer Literacy Goals for 1985 is but one evidence of this. Granted that defining computer literacy suffers from all the semantic problems of defining literacy, it would seem that the goal or ideal would be the ability to write one's own programs as well as to comprehend the social and political implications of the computer on people and societies thoughout the world. Within the framework of this definition, there inevitably will be levels of literacy dependent on the needs of each individual. In a democratic society, we cannot leave this literacy to an elite.

As one considers the issue of computer literacy, CAI and basic skills students it appears that among the major obstacles are faculty attitudes and competences relating to educational computing. Most faculty lack acquaintance with the computer and have not utilized it in their training. Where faculty have used computers in their graduate programs, it has been primarily in the area of statistical analysis. Another significant fact is that programs training elementary and secondary teachers make little effort to teach the use of the computer as an educational tool. Since it is desirable to introduce the student to the computer as soon as possible, the lack of trained teachers on the elementary level constitutes a serious problem. In higher education, faculty often cite lack of equipment, lack of time, and lack of expertise as reason for not moving toward computer use. Many faculty have developed a degree of computer anxiety, and feel uncomfortable with the computers which they do not really understand. They often have anxieties about learning a computer language. The heavy use of the computer as a tool of analysis in the sciences and business, tends to reinforce the concept that computers are for those skilled in mathematics. The alphanumeric characteristics of the computer languages further reinforces this, as do the close ties that often have existed between emerging computer science departments and the mathematics and science departments. In addition, the persons teaching computer languages have not unnaturally come from a background in mathematics. This has often led to teaching the language as mathematics, or at least so it has appeared. This has often caused older faculty who have suffered from math anxiety to transfer it to the learning of the computer language. This anxiety is naturally increased when they come across students who are more knowledgeable about computers and program development than they are. Many of the older faculty hope they will be safely retired before the use of the computer becomes commonplace. Current trends in higher education may exacerbate the problem as the pool of students decreases and the effects of the tenure system prevents younger and computer oriented faculty from being hired.

Or faculty may feel the computer is a "gimmick" and that it is somehow improper to entice the student to learn through computer programs, seemingly forgetting that the important fact is whether or not the student learns through the experience. The delivery system is not the goal of learning but a means to it. In many institutions there is a lack of incentive for faculty to become involved in learning more about developing educational uses for the computer. If there is administrative support primarily for management uses for the computer, the faculty can get away without the computer. In addition, faculty often share the same phobias as students do and are afraid that somehow or other they will break the computer or damage it as they use it.

To what degree do faculty need to be computer literate? In many instances, a fruitful partnership can be developed between the academician who can develop the logic of the program and a student who knows the language. The problem here is the degree to which the faculty person can develop the flow chart and logic of the program. The faculty should also be aware of the societal impact of the computer and its implications. At a minimum, in a transitional phase, faculty need to overcome the anxieties they may have about computers so that their attitude is not a block to student attitudes toward computer use. To introduce CAI into a basic skills program without prior training for the instructors will foredoom it to failure.

In fairness to faculty, it should be noted that not all the onus should be placed on them. Faculty who are aware of the potential of the computer and are willing to implement CAI find that software is not available. They do not have adequate encouragement in the way of released time or facilities to develop programs locally. And it has been noted that locally developed programs, perhaps because of time constraints, tend not to move above the lowest level of cognitive skills into problem solving, well thought out simulations or any of the higher levels of learning. Furthermore, it is frustrating for a faculty member with a desire to utilize CAI, to find the software unavailable (even if there are funds for purchase), and the computer devoted mainly to administrative uses. An additional problem has emerged with the proliferation of hardware. If programs do exist, they are frequently not transportable and the faculty member lacks the expertise to make the necessary modifications. Thus, for faculty working with developmental students, the problems may appear to be insurmountable for all but a few who will perserve for the common good, often without much support.

Should basic skills students be introduced early to the use of the computer as an aid to teaching them language, reading and math literacy? Given the necessary equipment and software, if they have not developed competence, they should start at the beginning of their college careers to become familiar with using the computer as an instructional tool. In using CAI programs, they will develop an awareness of the computer and become accustomed to using it. Although much of the current interest in remediation seems to be in math programs, limiting it exclusively to such programs might be detrimental in that it could reinforce the idea that it is only a mathematical tool. Without question, computers in basic skills programs could develop a level of awareness and familiarity that would lead to higher levels of literacy. This would imply that the instruction in basic skills could well be accompanied by some discussion of the role of the computer in society and its potential. If we are to develop a computer literate society, the time thus spent would not be wasted. It might be desirable to teach the student how to develop the shortest and simplest of programs, probably in BASIC, giving him some insight in how the computer is "trained" to do what it does. In any case, the use of CAI in basic skills programs can provide the student with the skills of using a computer and can serve as a motivator for him to go on to the higher degree of literacy of being able to write a program. The opportunity must be provided for these students to go as far as they need to with the computer.

Since minority institutions generally enroll students who have had the fewest educational advantages and have been much less likely to have had any prior exposures to educational computing, the importance of providing these students with the road to computer literacy is extremely important. For some time, college will remain the place where these students are introduced to the computer. These institutions have contributed significantly to the upward mobility of their students, providing a nuturing atmosphere for their development. These colleges have also traditionally been short of inrestricted operational funds. Thus they have found themselves disadvantaged in many ways in the attempt to institute programs that will provide computer literacy for their students. Often they have no faculty who have sufficient expertise to develop an institutional program for educational computing or to write grants. And without successful grant proposals, they do not have sufficient operational funds to channel into computer hardware, software, and faculty training. So the institutions and groups in society who need it most, are at a disadvantage in developing programs and suffer a double poverty.

Often the college is led to develop grants on a piecemeal basis without any overall policy with regard to use of the computer. Frequently this means the computer is perceived as a tool of financial management and its instructional capabilities are overlooked. Or the job-induced demand for computer science majors, leads to the development of a program directed mainly at producing a few computer science majors. Or the requirements of private foundations and industry may encourage the development of proposals for specific programs, such as math, or data processing, etc. This ought not to be the primary or only goal of any institution, least of all of minority institutions and other colleges with large numbers of basic skills students. Rather the overarching goal should be the exposure of all students to computers in order to provide a broadly based constituency of graduates who are computer literate and who will be able to program their own home computers.

Institutional policy alone is not enough to accomplish these educational goals. Without a national policy on the goals of educational computing and computer literacy can go unachieved. If the computer revolution is part of a process as revolutionary as the printing press and one that is much more accelerated, the development of a national policy and the provision of funds for computer literacy programs is a vital necessity. Current evidence suggests that students graduating today from high schools and possibly from college, do not have anything that could be considered a minimal level of computer literacy. Further, computer experience is not equally distributed across social groups. Minorities, women, persons living in rural areas, as well as those living in the southeast are less likely to have had any experience with the computer. The magnitude of the problem is shown by the following figures. On the one hand, in 1977, 71% of minority institutions reported they had computing equipment as opposed to 70% of other institutions. The telling factor is that only one-fortieth or 3% of the total number of students having experience with computers were in minority schools.

Current trends in national politics, the move toward returning responsibility to the states for educational programs, raise distrubing questions about educational computing. Not that it will disappear, but rather that underprepared students with potential will be left behind because their paths to computer literacy are limited.

The issue and the debate may be a political as much as an educational one. If one is committed to opportunity for all and if the computer is on the threshold of changing education, consciousness needs to be raised among educators, policy makers, and the public, This effort needs to make it clear that money alone will not solve the problem. Sufficient funds to buy sophisticated equipment will not by itself solve the problem without a faculty with computer awareness and computer literacy. Time and funds for well thought out faculty training programs are necessary. Perhaps a programmer should be added to the staff to work in partnership with the faculty in developing local programs. For without software, neither equipment nor faculty training will be effective.

The debate needs to move from what might be called technical literature aimed at those interested in educational computing into the public arena. The potential for computer delivery systems to bypass the innately conservative institutions of higher education can only cause more problems for educators. And the lack of a national policy and the action to achieve a computer literate society which would leave the disadvantaged even more disadvantaged is a course neither higher education nor the nation can afford. There are no simple answers.

## REFERENCES

Austing, Richard H. A Study of Computer Impact on Society and Computer Literacy Courses and Materials. Journal of Educational Technology Systems, 7(1978-79), 267-273.

Anderson, Ronald E. National Computer Literacy, 1980. Unpublished paper presented December 1980 at National Computer Literacy Goals for 1985 Conference in Reston, Virginia.

Bork, Alfred and Franklin, Stephen D. The Role of Personal Computer Systems in Education. AEDS Journal, 13(Fall, 1979), 17-30.

Caldwell, Robert M. Guidelines for Developing Basic Skills Instructional Materials for Use with Microcomputer Technology. Educational Technology, December, 1980, 7-12.

Diem, Richard A. and Fairweather, Peter G. An Evaluation of A Computer-Assisted Education System in An Untraditional Academic Setting - A County Jail. AEDS Journal, 13(Spring, 1980), 204-213.

Jay, Timothy B. Computerphobia: What To Do About It. Educational Technology, 21(January, 1981), 47-48.

Johnson, D. C., Anderson, R. E., Hausen, T. P., Klassen, D. L. "Computer Literacy - What Is It? Mathematics Teacher, 73(February, 1980), 91-96.

Diem, Richard A. "Developing Computer Skills: An Inservice Training Program". Educational Technology, February, 1981, 30-32.

Johnson, James W. Getting from Here to There: The States of Instructional Computing in Higher Education, T.H.E. Journal, 7(November, 1980), 48-53.

Lautsch, John C. The Computer and Education: The Genie is Out of the Bottle, T.H.E. Journal. 8(February, 1981), 34-35, 39.

Lower, Stephen E. Making C.A.I. Make A Difference in College Teaching, ERIC Document, Ed129 299.

McIsaac, Donald N. Impact of Personal Computing on Education. AEDS Journal, 13(Fall, 1979), 7-15.

Marshall, Sister Patricia Instructional Computing in Minority Institutions, A Needs/Strategy Assessment. Executive Summary of Final Report/ NSF SPI 7821515, 1980.

Minnesota. Educational Computing Consortium. Highlight Report. A Study of Computer Use and Literacy in Science Education. August, 1980.

Molnar, Andrew R. The Next Great Crisis in American Education: Computer Literacy. Journal of Educational Technology Systems, 7(1978-79), 275-284.

Morsund, David Microcomputers Will Not Solve The Computers-in-Education Problem, AEDS Journal, 13(Fall, 1979), 31-39.

Seidel, Robert L. It's 1980: Do You Know Where Your Computer Is? Phi Delta Kappan, March, 1980, 481-485.

Shostak, Arthur B. The Coming Systems Break: Technology and Schools of the Future. Phi Delta Kappan, 62(January, 1981), 356-359.

Stevens, Dorothy J. How Educators Perceive Computers in the Classroom. AEDS Journal, 13(Spring, 1980), 221-232.

## INDEX OF PAPERS BY TOPIC

### ADMINISTRATIVE COMPUTING

| | | |
|---|---|---|
| Alspaugh, John W. | Computer Assisted School Bus Route Construction With A Microcomputer | 1 |
| Bailey, T.E. | Pseudocode And Constructs For Structured Programming In FORTRAN (1966 Standard) And BASIC | 3 |
| Berry, Robert I. | Zero Base Budgeting A Management System Approach For Educational Institutions | 18 |
| Carruth, Gayden F. | Position Paper: Common Information Base For Regional And Local Decision Making | 52 |
| Daniels, Arthur W. | The Development And Operation Of A Programming Cooperative | 69 |
| Delf, Robert M. | Application Software: Buyer's Guidelines | 74 |
| Gockel, James | Large Time-Sharing Microcomputers In Secondary Education | 100 |
| Hadduck, Charles G. | An Employment Applications Information System | 107 |
| Haugo, John E. | Management Applications Of The Microcomputer: Promises and Pitfalls | 129 |
| Kahn, Phyllis | The State's Role In Educational Computing: View From The Legislature | 157 |
| LaChance, Doug,<br>Stokka, Keith | Using A Microcomputer With A Large Data Base | 169 |
| Majors, A.V.,<br>Westerfeld, Lloyd | Student Intelligent Terminal System | 190 |
| Mayer, Herbert C. | Information Systems Management | 195 |
| Raucher, Stephen M.,<br>Koehler, Tommas J. | Management Operations Review And Evaluation (MORE) Of A School District Computer Services Department | 217 |
| Settlemire, James R. | Distributed Systems In The School District Environment | 253 |
| Weitz, Russell E. | Word Processing Evaluation And Selection | 275 |

COMPUTER AWARENESS/LITERACY

| | | |
|---|---|---|
| Bell, Fred | Computer Literacy And Computer Uses In Education: What Should They Be? | 12 |
| Nelson, Robert W.,<br>Kepner, Henry S. | Computers In Education - Meeting The Challenge | 210 |
| Postman, Robert D. | ROBOT PROBE - An Introduction To Programming | 214 |
| Rice, Jean,<br>O'Connor, Sandy | Computer Literacy In The Classroom | 229 |
| Rich, Barbara | Implementation Of A College-Wide Computer Literacy Program: A Working Model | 234 |
| Wachtel, Edward E. | Computer Science Summer School Offering - A Place To Start | 267 |

COMPUTER LANGUAGES

| | | |
|---|---|---|
| D'Angelo, John | TI LOGO: A Turtle Of A Different Color | 65 |

COMPUTERS AND MUSIC EDUCATION

| | | |
|---|---|---|
| Blombach, Ann K.,<br>Murphy, Barbara A. | Music Ear Training At Ohio State: The Phoenix/Orion Project | 29 |
| Gross, Dorothy,<br>Griffin, Wendy | Implementation And Evaluation Of A Computer-Assisted Course In Musical Aural Skills | 104 |

COMPUTERS AND SPECIAL EDUCATION

| | | |
|---|---|---|
| Erting, Lynne,<br>Mackall, Phil | Academic And Non-Academic Benefits From Using PLATO - A Classroom Teacher's View | 89 |
| Garvey, Mary | Use of CAI In A Mainstreamed Hearing Impaired Program | 97 |
| Koehler, Tommas J.,<br>Ossler, Anna C.,<br>Raucher, Stephen M. | Computerized Support Systems For Individualized Educational Plans | 165 |
| Loewen, Duane | Microcomputers In Teaching Handicapped Students | 182 |
| Lorton, Paul,<br>Muscat, Eugene J. | Administrative Use Of The Thimble: A Micro-Computer Attendance System | 185 |
| Whitney, Richard A.,<br>Hofmeister, Alan M. | MONITOR: A Computer Based Management Information System For Special Education | 279 |
| Wilde, Jane H. | The Computer In Developmental Education | 290 |

## COMPUTERS AND THE MEDIA CENTER

| | | |
|---|---|---|
| Carrison, Dale K. | MSUS/PALS: Planning For The Future | 48 |

## COMPUTERS IN HIGHER EDUCATION STRAND

| | | |
|---|---|---|
| Hassell, Johnette | A Paradigm For Developing A Computer Science Program | 125 |
| Mosow, David K. | The Microcomputer and Management of the Time Bound Educational Program | 206 |
| Roll, Peter | Long Range Planning For Information Services At Large Universities | 248 |
| Sinofsky, Esther R.,<br>Knirk, Frederick G. | A Systematic Procedure For Problem Analysis And Proposal Development | 257 |

## INSTRUCTIONAL COMPUTING

| | | |
|---|---|---|
| Barber, Frank | People Make The Difference | 9 |
| Boe, Thomas F. | Effective Integration Of Computer Methods In The Science Laboratory | 34 |
| Borry Hausmann, Linda | Effective Feedback In Computer Assisted Instruction | 38 |
| Brebner, Ann,<br>Hallworth, H.J.,<br>Woetowich, H.M.,<br>Mah, S.N.,<br>Huang, H.C. | Results And Implications Of Some Recent Research On CAI In Mathematics | 42 |
| Browne, Diane,<br>Corens, Ken | Microcomputers In Cooperative Groups | 47 |
| Dalgaard, Bruce R.<br>Askalani, Mohamed<br>Thompson, Paul | Computer Assisted Instruction in Economics: Development, Results And Challenges | 59 |
| Dellow, Donald A.,<br>Poole, Lawrence, H.,<br>Clark, Forrest | The Role of Microcomputers In Community College Business And Secretarial Science Programs | 80 |
| Donoian, George | A Paradigm For Computer-Based Instruction: Renaissance Of The Affective Domain | 84 |
| Heck, Bill | Microcomputers In Education | 133 |
| Hurly, Paul<br>Davis, Virginia | Lifelong Learning And Post-Industrial Revolution | 139 |

| | | |
|---|---|---|
| Isaacs, Gerald L. | Realities Of Transportable Micro-Courseware With Graphics Emphasis | 145 |
| King, Gary,<br>Kobylarz, Linda | The GIS Options Program: A Dynamic Approach To Providing Localized Career Information | 162 |
| Lindsay, E. Jay | Learning By Computer Practice | 176 |
| Molloy, Lawrence,<br>Herstein, Ed | A Computer Demonstration Room For Educators | 203 |
| Rawitsch, Don G. | The Organization Of User Services For Instructional Computing | 222 |
| Rizza, Peter J. | For Computer-Based Education To Matter It Needs Critical Mass | 237 |
| Wachtel, Edward E.,<br>Rankin, H. Clair,<br>Obrock, Madeline,<br>Burns, Evelyn,<br>Bazeley, John | A Five Year Time Line For The Integration Of Computers Into the Shaker Heights Schools | 270 |
| Wiederanders, Donald E.,<br>Schwandt, Lynn C. | Microcomputers: Integration Into The Curriculum | 287 |

## INSTRUCTIONAL DESIGN

| | | |
|---|---|---|
| Hartman, Janet,<br>Garnett, Dianne | A Structured Approach To The Development Of Computer Assisted Instruction | 119 |
| Roblyer, M.D. | Instructional Design vs. Authoring of Courseware: Some Crucial Differences | 243 |

## INSTRUCTIONAL MANAGEMENT

| | | |
|---|---|---|
| Cogdill, Linda J.,<br>Goldberg, Albert L. | An Information System For Managing HEAD START | 56 |
| Hamilton, Gregory C. | A Microcomputer Managed Instructional System | 112 |
| Uffelman, Robert L.,<br>Boas, Edward E. | Delta Computer-Managed Instruction System | 260 |

## SOFTWARE CLEARINGHOUSES/EVALUATION

| | | |
|---|---|---|
| Holznagel, Donald C. | A Clearinghouse For Educational Applications Of Microcomputers | 136 |
| Judd, Dorothy H. | Sources For And Evaluation Of K-12 Microcomputer Software | 150 |
| Lathrop, Ann | Softswap: The Evaluation And Dissemination Of Non-Commercial Teacher Created Software | 173 |
| Van Dusseldorp, Ralph | Alaska On-Line Resource Information Bank | 263 |

## VIDEODISC

| | | |
|---|---|---|
| Kehrberg, Kent T.,<br>Pollak, Richard A. | Low Cost Computer/Videodisc Learning Stations | 159 |